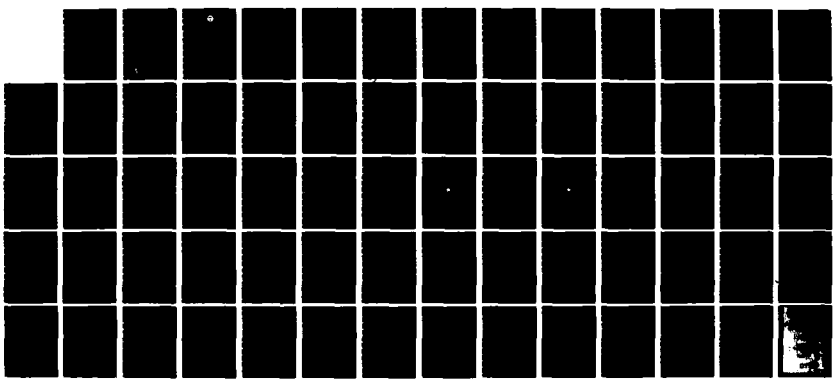
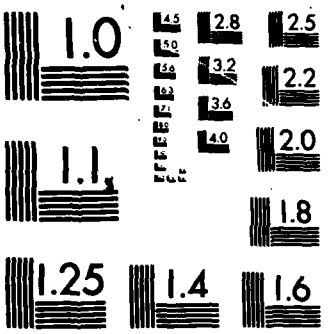
AD-A137 552 LIU FAULT DIAGNOSIS INVESTIGATION(U) WESTEC SERVICES INC SAN DIEGO CA K J CAMPBELL ET AL. NOV 83 NOSC-TD-626 N00123-81-D-0437 1/1

UNCLASSIFIED F/G 9/2 NL





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

(12)

NOSC TD 626

NOSC TD 626

AD A137552

Technical Document 626

LIU FAULT DIAGNOSIS INVESTIGATION

K. J. Campbell (NOSC)
P. L. Stamm
(WESTEC Services, Inc.)
Contract N00123-81-D-0437

November 1983
Final Report
January — June 1983

Prepared for
Naval Electronic Systems Command

Approved for public release; distribution unlimited.

DTIC FILE COPY

NOSC

NAVAL OCEAN SYSTEMS CENTER
San Diego, California 92152

NAVAL OCEAN SYSTEMS CENTER SAN DIEGO, CA 92152

AN ACTIVITY OF THE NAVAL MATERIAL COMMAND

J.M. PATTON, CAPT, USN
Commander

R.M. HILLYER
Technical Director

ADMINISTRATIVE INFORMATION

This work was performed under Naval Ocean Systems Center (NOSC) problem EF23 and was sponsored by the Naval Electronic Systems Command (NAVELEX), Code 614. The work described herein was performed by WESTEC Services, Inc. under contract N00123-81-D-0437 under the direction of the Command and Control Applications Branch, Code 8254, NOSC.

Released by
G.E. Ereckson, Head
Information Transfer and
EW Division

Under authority of
J.H. Maynard, Head
Command Control-Electronic Warfare
Systems and Technology Department

LO

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER NOSC Technical Document 626 (TD 626)	2. GOVT ACCESSION NO. AD-A137 552	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) LIU FAULT DIAGNOSIS INVESTIGATION		5. TYPE OF REPORT & PERIOD COVERED Final January–June 1983
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) K.J. Campbell (NOSC) P.L. Stamm (WESTEC Services, Inc.)		8. CONTRACT OR GRANT NUMBER(s) N00123-81-D-0437
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Ocean Systems Center San Diego, CA 92152		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS EE23
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Electronic Systems Command Washington, DC 20360		12. REPORT DATE November 1983
		13. NUMBER OF PAGES
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Fault diagnosis
Fault detection
dc circuits
Fault resistance

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This report is an investigation of the fault diagnosis algorithm attributed to Dr. Ruey-wen Liu at the University of Notre Dame. This approach is based on a fault current concept in which a faulty component is modelled as a nominal component with a parallel fault current source.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-LF-014-6601

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

EXECUTIVE SUMMARY

This report is an investigation of the fault diagnosis algorithm attributed to Dr. Ruey-wen Liu at the University of Notre Dame. This approach is based on a faulty current concept in which a faulty component is modelled as a nominal component with a parallel fault current source. Single component faults trace a line in a measurement space as the fault current source takes on different values. Measurements taken on a sample circuit form a point in the measurement space. The origin of the measurement space is the intersection of the fault lines of the various circuit components and represents the circuit with all components at their nominal values. The Liu fault diagnosis approach is to determine which fault line is closest to the measurement point of the sample circuit.

The circuit used to evaluate the Liu approach is an eight-resistor bridged-T dc network. This nonreactive circuit has been chosen to keep the results simple enough to provide insight into the approach. The dc circuit has two ports, each driven by fixed current sources. The two-port voltages are the coordinates of the two-dimensional measurement space. The effect of component tolerance was taken into consideration when evaluating each of the eight single component faults. A 5-percent worst case tolerance situation was simulated for each fault. In this simulation, components other than the fault itself are set to either +5 percent tolerance or -5 percent tolerance. The combination of tolerances producing the most severe deviation in the output port voltages is chosen as the worst case situation. The fault resistance value is allowed to range from 0 to infinity and the worst case point is calculated.

The results of the investigation revealed that these points outline a trapezoidal region representing the 5-percent worst case tolerance for the entire range of each single component fault. The regions outlined for the eight components overlap considerably near the origin of the measurement space making diagnosis difficult. However, when the faults become extreme the regions separate, making fault diagnosis feasible.

The Lui approach has the advantage of allowing a continuous range of faulty component values. As such, specific fault values are not required to be prehypothesized. By comparison the traditional nearest neighbor rule which uses a fault dictionary approach, has only discrete fault conditions as potential fault candidates for diagnosis.

CONTENTS

FIGURES

FIGURES (Continued)

TABLES

SECTION 1

BACKGROUND

In an attempt to provide methods of fault detection and diagnosis in analog circuits, various universities have participated in an effort sponsored by the Office of Naval Research directed towards the development of new mathematical tools. One of these efforts was directed by Dr. Ruey-wen Liu of the University of Notre Dame. This report investigates the practical aspects of implementation of this approach and evaluates its ability to correctly identify known faults.

SECTION 2

CANDIDATE DC CIRCUIT

SELECTION OF THE CANDIDATE DC CIRCUIT

The circuit selected for evaluation of the Liu approach to fault diagnosis is a bridged-T resistive network. The circuit as shown in Figure 1 contains eight resistive components. The components are labelled as conductances G4 through G11. Conductance is the reciprocal of resistance and is used because it is more natural for the fault current model which is central to the Lui approach. This is covered in detail in Section 3. The conductances of all the components except G11 are 1 mho. Component G11 has a conductance of 2 mho. These values are listed in Table 1. The circuit is driven by two current sources I_1 and I_2 driving the ports. Voltages V_1 and V_2 are measured across the two output ports. For the purposes of this evaluation the current sources are either symmetric $I_1 = I_2 = 1$ ampere or asymmetric $I_1 = 1$ and $I_2 = 2$ amperes. The circuit has differing behavior under these two conditions.

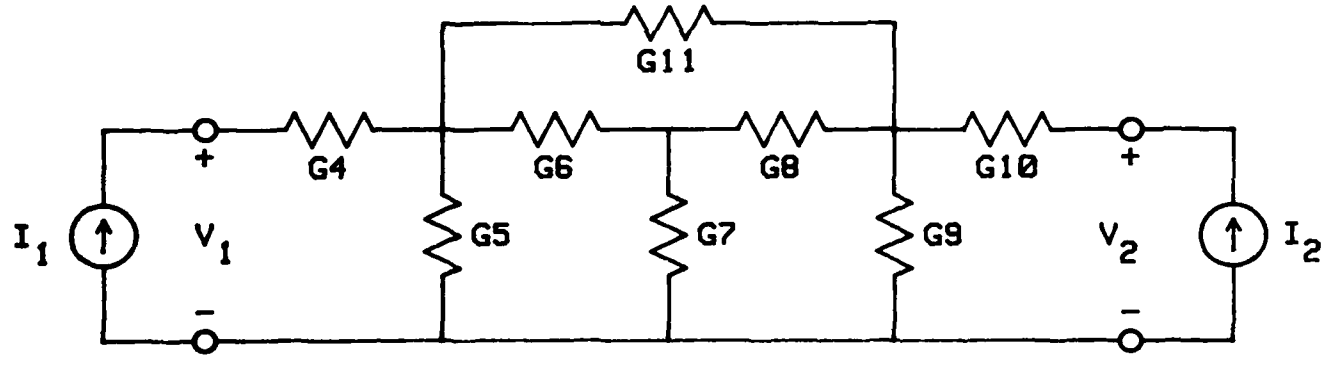


Figure 1. Candidate dc circuit.

The measurements of V_1 and V_2 discussed in this report are incremental changes of V_1 and V_2 away from the quiescent state in which all components are nominal. The incremental changes are caused by component variation due to either a single component fault or nonfaulty component tolerance or both. The term component tolerance is used to describe the variation of nonfaulty components.

Table 1. Nominal component values for dc circuit.

Component Number	Nominal Resistance, ohm	Nominal Conductance, mho
4	R4 = 1.0	G4 = 1
5	R5 = 1.0	G5 = 1
6	R6 = 1.0	G6 = 1
7	R7 = 1.0	G7 = 1
8	R8 = 1.0	G8 = 1
9	R9 = 1.0	G9 = 1
10	R10 = 1.0	G10 = 1
11	R11 = 0.5	G11 = 2

This dc circuit was chosen as the core of our evaluation because it is straightforward enough to provide insight into the Liu approach to fault diagnosis. This approach adds fault current sources in parallel with each branch. Even this circuit has a large number of variables associated with it. Two output ports and two measurement quantities limit the problem to two dimensions, which makes for greater insight into the approach. The concept may readily be extended to multidimensions but is much more difficult to visualize. A non-reactive circuit allows the measurements to be at one frequency (dc) and the measurements are not complex quantities. The approach may readily be extended to multiple frequencies.

SOLUTION OF THE CANDIDATE DC CIRCUIT

The candidate dc circuit is straightforward but the addition of a fault current source in parallel with each component complicates the solution of the circuit. The circuit is driven by current sources I_1 and I_2 and the output port voltages V_1 and V_2 are the measured quantities. Using the fault current model from the Liu approach to fault diagnosis, there are eight component currents, eight component voltages, and eight fault currents to be considered. It is necessary to carefully weigh which of these variables are dependent and which are independent. The desired solution, given the component values, which may deviate from nominal, and the driving currents I_1 and I_2, is the output port voltages V_1 and V_2 and the slopes of the fault lines. The meaning of these slopes is discussed in Section 3.

The circuit is solved using Kirchoff's Voltage Law and Kirchoff's Current Law. The resultant equations form a 10 x 10 matrix. This matrix must be inverted to obtain the two output voltages and eight fault line slopes.

SECTION 3

LIU APPROACH TO FAULT DIAGNOSIS

INTRODUCTION

The Liu fault dianosis approach is based on the fault current concept. A faulty component is modelled by a nominal component with a parallel fault current source. This current source produces exactly the same effect on the circuit as the component becoming faulty.

Measurements of currents or voltages within the circuit or at external ports form a multidimensional space in which each variable represents a dimension. In a linear circuit the injection of fault currents linearly adds to the measurements in proportion to the fault currents. The locus of points generated by the fault current taking on a range of values is a straight line. In a nonlinear circuit the locus of points is also a line but it is not necessarily straight or even continuous. These lines are associated with each single component fault and are termed fault patterns.

The Liu approach also considers the question of separability of the fault patterns. If the fault lines lie close together it is difficult to distinguish one from the other. A fault is diagnosed by making measurements on the sample circuit which determine a point in the multidimensional measurement space. The fault line lying nearest, by some criterion, is diagnosed as the fault in the circuit.

LIU FAULT PATTERN APPROACH

The concept of fault currents is fundamental to the Liu fault diagnosis approach. Figure 2 is an illustration of the fault current concept. The nominal component or branch has a complex admittance Y which is the reciprocal of its complex impedance Z. This branch may be purely resistive, capacitive, or inductive or a combination for purposes of the model. Admittances are appropriate to illustrate the fault current concept. When the nominal component becomes faulty it may be modelled by an increment in admittance ΔY which is parallel with Y. Admittances in parallel add so the faulty component is modelled by an admittance $Y+\Delta Y$. The incremental admittance may be replaced by an appropriate current source J which mimics the behavior of the circuit with a fault component. The sign and magnitude of J are such as to accomplish this replacement. With an applied voltage V across the branch, the fault current is $J = V \cdot \Delta Y$. This current source is also referred to as the fault compensator for this branch because it compensates the nominal component to appear as a faulty one. In general J, V, and ΔY may be complex quantities. A branch that is purely resistive, as is the candidate dc circuit, has a purely real admittance symbolized by the conductance G, the reciprocal of the resistance. In the resistive case, then, $J = V \cdot \Delta G$.

GENERATION OF FAULT PATTERNS

One of the major consequences of the fault current model is particularly interesting for linear circuits. The concept may be equally well applied to

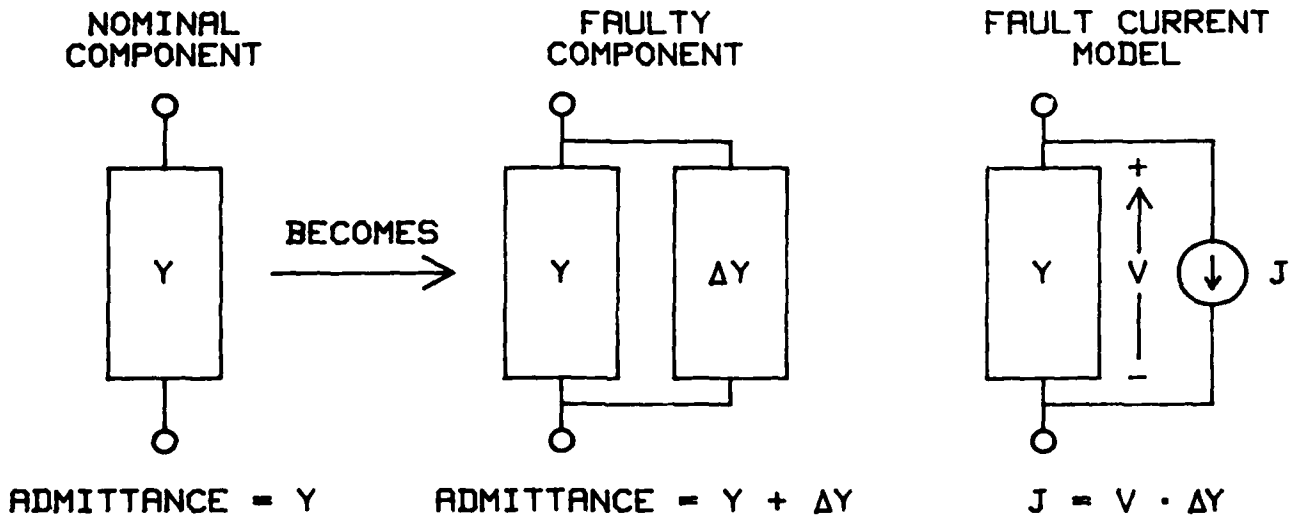


Figure 2. Fault current concept.

nonlinear circuits. In a linear circuit the superposition theorem applies. This theorem states:

> The response to several independent current or voltage sources is the sum of the responses to each independent source with the remaining sources zeroed (dead).

The consequence is that the measured values of currents or voltages in the circuit are linearly proportional to the fault current. Some measurements may be independent of fault currents or very slightly affected by them. This is particularly likely when the fault current source and the measurement are distant from each other in the network. The linear relationship of the measurements and fault currents implies that the locus of points traced in the multidimensional measurement space is necessarily a straight line. The fault patterns for the candidate dc circuit are shown in Figure 3. The patterns consist of straight lines of various slopes passing through the origin. The origin in V_1, V_2 space is actually the port voltages measured with all nominal components and no faults. In the figure, V_1 and V_2 are actually increments away from the nominal circuit. These lines do not take into account the fact that with physical components there are limits on the lines. The physically limited locus of points on the fault line is determined by the faulty component taking on values of R=0 (short) and R=infinity (open).

The physically limited fault patterns for the symmetric current source case I_1 = 1 ampere, I_2 = 1 ampere are illustrated by Figure 4. The physically limited fault patterns for the asymmetric current source case I_1 = 1 ampere, I_2 = 2 amperes are illustrated by Figure 5. These are illustrations of two distinct conditions and do not exhaust all possible current source values. The increase in source current for I_2 to 2 amperes in the asymmetric case increases the length of the fault lines but does not change their slope. This is to be expected from the superposition theorem for a linear circuit. The

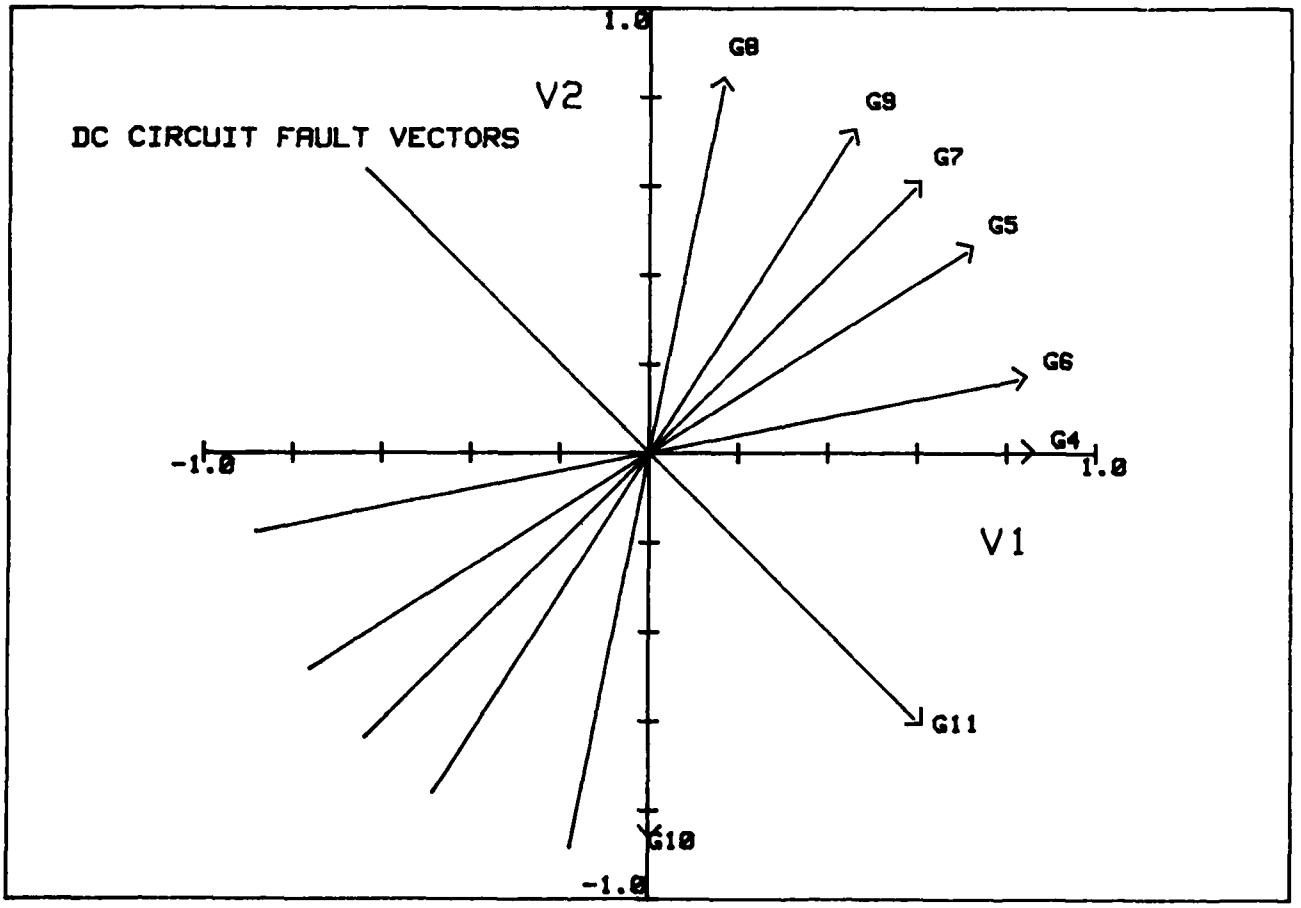


Figure 3. DC circuit fault vectors.

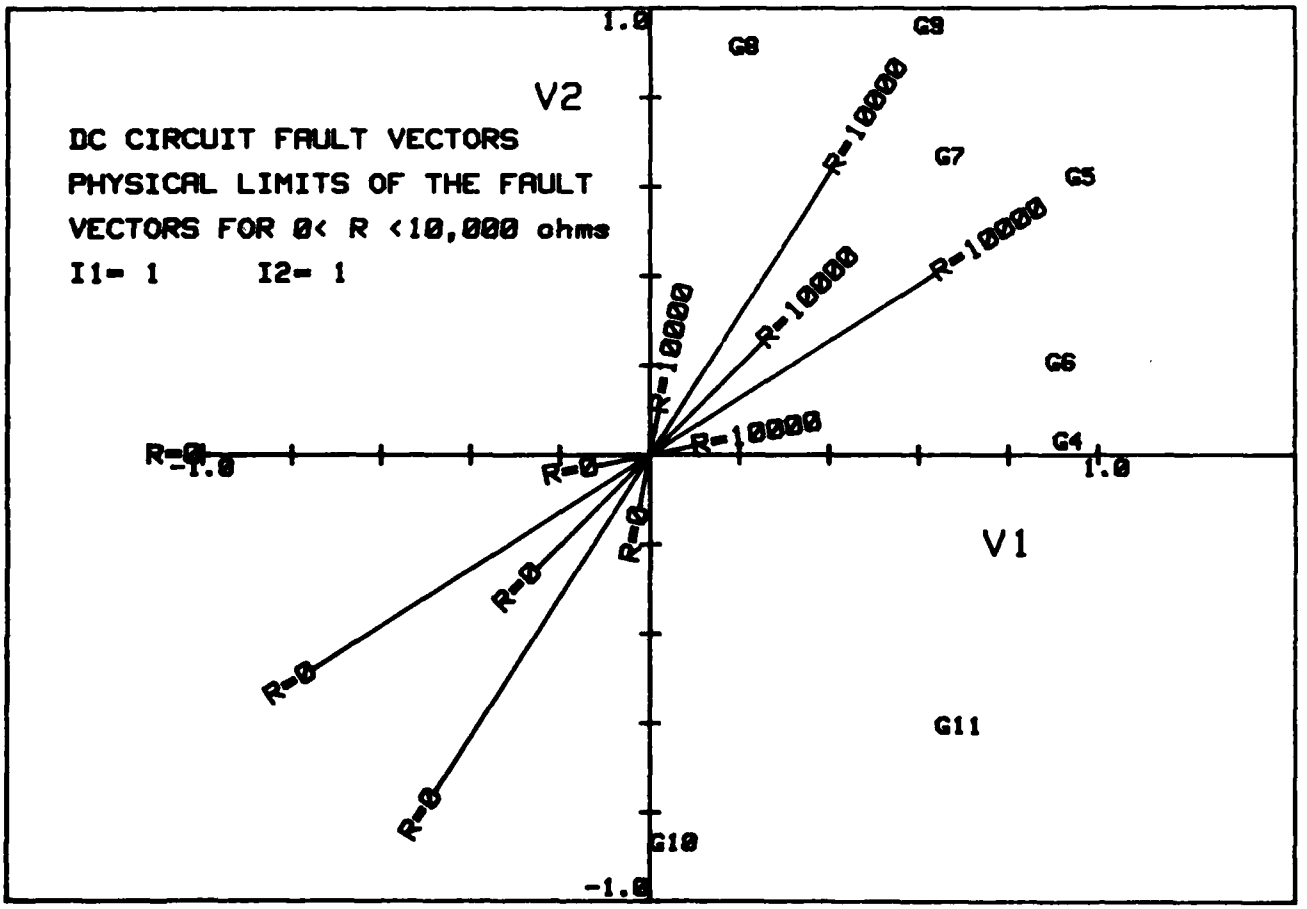


Figure 4. DC circuit fault vectors with physical limits (symmetric current sources).

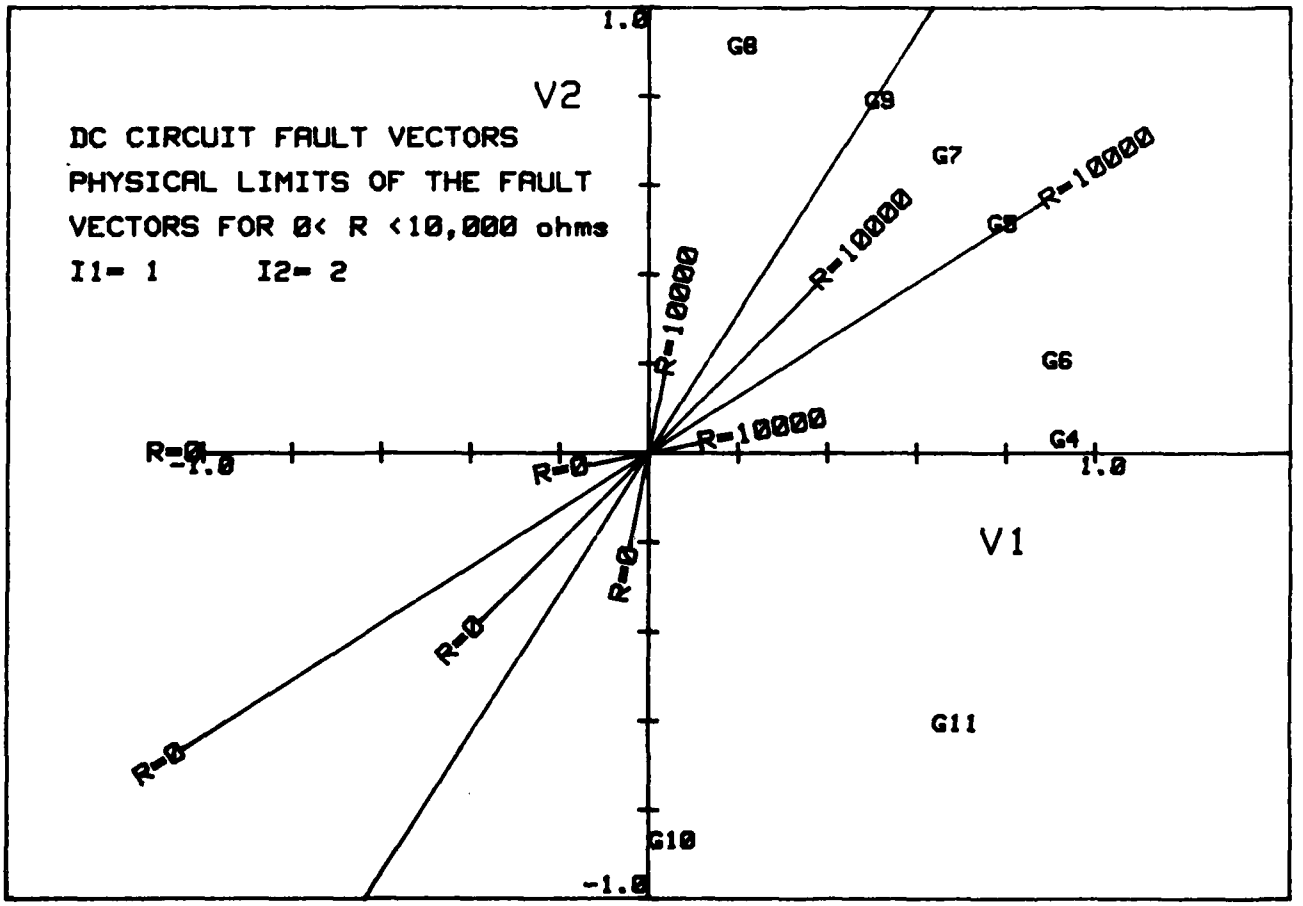


Figure 5. DC circuit fault vectors with physical limits (asymmetric current sources).

fault lines are labelled at each end with R=0 (short) and R=10,000 (effectively an open). Any larger value for R would not show on the illustration. The fault lines for input conductances G4 and G11 go to large values of V_1 and V_2 off the plot because they are in series with current sources I_1 and I_2, respectively.

SEPARABILITY OF FAULT PATTERNS

The pattern associated with single component faults is a fault line in multidimensional space. Fault lines may have very similar slopes and cluster near each other. This is a problem particularly plaguing large, complex circuits. Theoretically, faults may be correctly diagnosed no matter how small the angular separation of the fault lines. This is because a single component fault, with the remaining component perfectly nominal, will produce a measurement falling exactly on the appropriate fault line. In practice even a small amount of component tolerance may change the slope of many of the fault lines. This is discussed in more detail in Section 4.

EFFECT OF COMPONENT TOLERANCE

Figure 6 illustrates the effect of component tolerance on the fault line (or vector) for component 6. The line (or vector) labelled G6 is that obtained with all nominal component values with the exception of single component faults in component 6. To illustrate the effect of component tolerance, the line (or vector) labelled G6' which lies above the G6 line is obtained with all nominal component values with the exception of R7, which is 1.3 ohms rather than 1 ohm, and the exception of single component faults in component 6. The effect is to both shift the fault line and alter its slope. The origin shift point on Figure 6 is labelled with an "*." This is the point in measurement space where component 6 takes on its nominal value of 1 ohm. The origin has been shifted due to a tolerance of +30 percent in R7. In addition, the slope of the component 6 fault line has become slightly smaller due to the effect of +30 percent tolerance in R7 on the whole network.

DECISION CRITERIA

For the ideal case where all components are exactly nominal with the exception of the single fault, that fault would be diagnosed as belonging to the correct fault pattern. An exception would be two fault lines identical in slope. The decision criterion applied to determine which fault pattern is "closest" to the measurement may affect which fault is diagnosed. Choosing the fault pattern lying the closest in the Euclidean sense to the measurement point is the nearest neighbor criterion. Choosing the most likely fault pattern is the maximum likelihood criterion. The maximum likelihood decision criterion takes into account the probability distribution of each component's tolerance and the probability distribution of each fault. The maximum likelihood criterion weighs the Euclidean distance measure by the inverse of the standard deviation of the associated probability distribution. The closest in this sense is the most likely fault. The difficulty with this method is that a probability density or at least a standard deviation must be known or assumed for each component. If the standard deviations associated with the components are assumed to be equal, then the maximum likelihood and nearest neighbor criteria become the same.

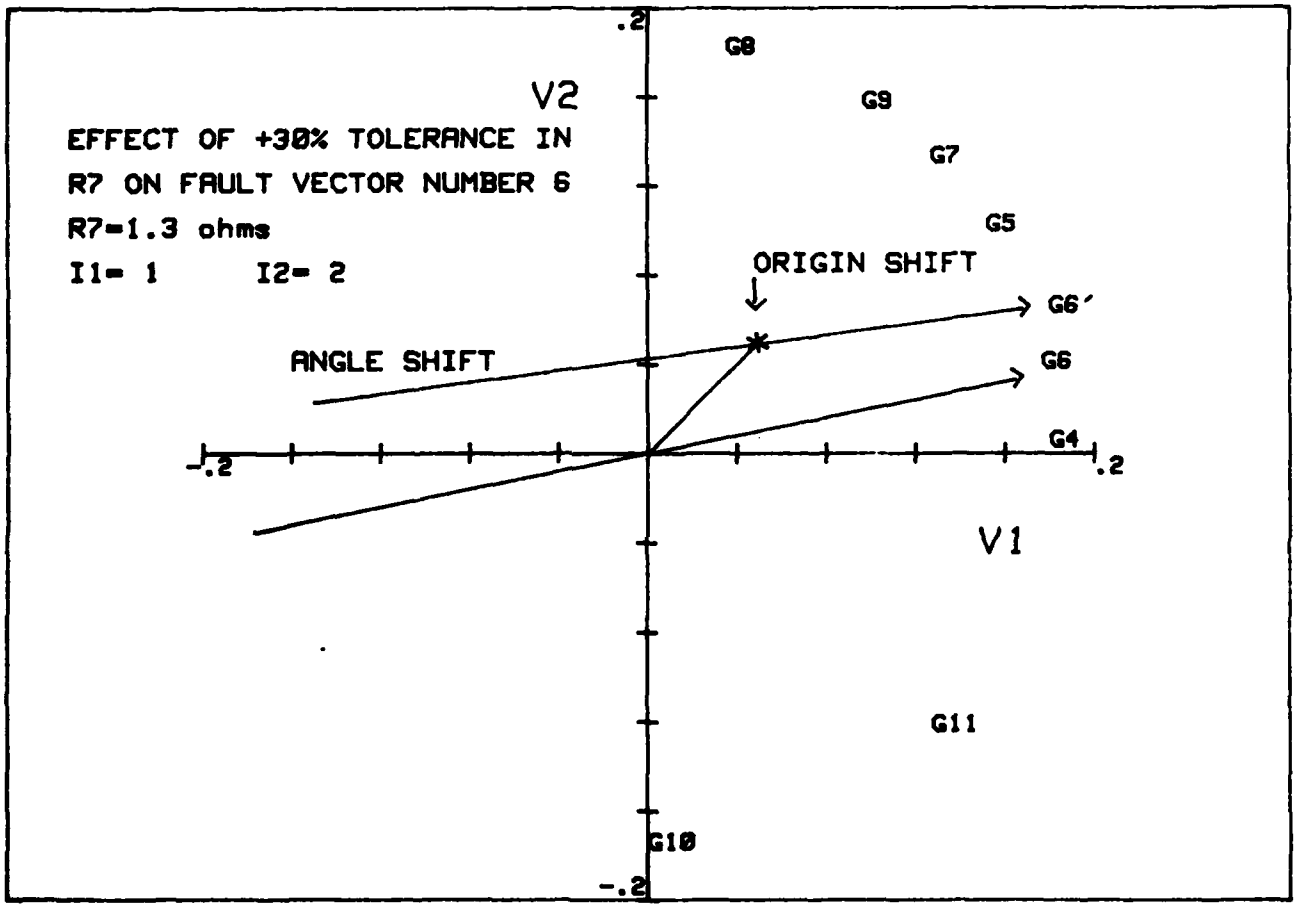


Figure 6. Vector angle and origin shift due to component tolerance.

ROBUSTNESS CONSIDERATION

The robustness of a fault-diagnosing scheme is its immunity to component tolerance in the nonfaulty components. A lack of robustness is a major shortcoming of most fault-diagnosing schemes. This is because actual circuits do have variations in their component values which cause the fault patterns to shift. Fault patterns which cluster near each other without component tolerance become indistinguishable in the presence of component tolerance. Fault patterns which have distinctly different slopes are more likely to be distinguished in the presence of component tolerance. The greater the extent of component tolerance the more likely faults may not be distinguished and properly diagnosed. The greater the severity of a fault the more likely it is to be correctly diagnosed.

SECTION 4

EFFECT OF COMPONENT TOLERANCE ON THE DC CANDIDATE CIRCUIT

INTRODUCTION

The real challenge to a fault-diagnosing approach is component tolerance. This is variation of the nonfaulty components away from nominal design values. For this evaluation the faulty component is allowed to range through a set of physically attainable values. The remaining components are allowed to vary +5 percent or -5 percent as they would off-the-shelf. The reason that component tolerance is a challenge to the diagnosis approach lies in the pattern matching, which is the basis of the approach. Measurements made on the sample circuit produce a point in the measurement space. This point is matched to one of the fault patterns. With no component tolerance this measurement point would fall exactly on a fault pattern. Fault diagnosis is straightforward with no tolerance. Component tolerance moves this point away from the correct fault pattern and possibly toward another fault pattern. The nearest fault pattern may be the wrong choice. The greater the component tolerance, the more likely the fault diagnosis will be incorrect.

WORST-CASE 5-PERCENT TOLERANCE

To determine the robustness of the Lui fault diagnosis approach a worst-case analysis was implemented. This was done by computer simulation on the Hewlett-Packard HP9826 computer and associated printer and plotter. Worst-case fault region plots drawn on the HP plotter are presented as Appendix A. This hardware system is described in Appendix B. The software is listed in Appendix C. The software is described later in this section. Component tolerance of 5 percent above and below nominal was chosen as representative of the extreme state of tolerance for each nonfaulty component. These are the upper and lower limits for 5 percent components typically found in practical circuitry.

Single component faults were simulated for each of the eight components G4 through G11. The faulty component was allowed to take on values from 0 to 50 ohms. Table 2 lists the set of values used to produce the fault region plots to analyze robustness to tolerance. The resistance values are chosen to produce relatively even increments along the fault line. They are not intended to be uniform steps of resistance. The simulation illustrates the conditions in which the driving current sources are the same I_1 = 1 ampere and I_2 = 1 ampere, and dissimilar I_1 = 1 ampere and I_2 = 2 amperes. These current source conditions are termed the symmetric and asymmetric cases, respectively.

The worst-case region for each component and for the symmetric and asymmetric current cases was determined by computer simulation. These regions for each fault type are discussed in detail in Section 4.5. The regions are traced by the voltages V_1 and V_2 computed for the set of faulty resistance values while the remaining components take on the worst possible set of +5 percent or -5 percent tolerances. The worst set of tolerances will have some components at +5 percent and others at -5 percent. The set which causes the calculated V_1, V_2 point to be the furthest from the fault line is the worst-case set.

Table 2. Fault resistance range.

Resistance (ohms)	Condition Simulated
0.00	Short
0.05	Low Value
0.10	Low Value
0.20	Low Value
0.35	Low Value
0.50	Nominal (R11 only)
0.70	Low (except R11) High (R11 only)
1.00	Nominal (except R11)
1.50	High Value
2.00	High Value
3.50	High Value
10.00	High Value
50.00	Open

WORST-CASE SIMULATION METHOD

The worst case of component tolerance is that combination of 5-percent tolerance extremes above and below nominal in the seven nonfaulty components which results in the greatest displacement of V_1 and V_2. The displacement is from V_1 and V_2 measurements with the nonfaulty components at their exact nominal value. There are $128 = 2^7$ such combinations that must be considered to determine the worst possible combination.

A rigorous network solution for all of these 128 combinations and for each of 8 possible faulty components and each of 13 fault resistance values is a lengthy calculation. There are a total of 13,312 such cases requiring a network solution. The HP9826 is extremely fast, but the dc circuit solution requires 4.58 seconds to determine the port voltages. Performing the simulation directly would require 1016 minutes, which is nearly 17 hours. This calculation time is for just the symmetric current source case and would be repeated for the asymmetric current case. While this is not beyond reason by itself, there is also the interaction of the HP9872B plotter which takes several minutes for each fault region. Bearing this in mind, the simulation would require 3 working days for each current source condition. The plots have individual artifacts which require running the entire program to discover

and correct. This and the general problem of perfecting the program make a 3-day turnaround impractical. A linearization technique based on perturbation theory is an answer to this problem.

The network solution centers around the 10 x 10 connection matrix Q which is formed in the program DC CIRCUIT from component values in the network. The values in the Q matrix are actually component values and their location in the Q matrix is determined by the connection topology of the network. A rigorous solution of the network requires that Q be inverted. The actual output port voltages V_1 and V_2 are linearly formed from the input currents I_1 and I_2 and Q^{-1} which is the inverse of Q. Small changes in component values are reflected in small changes in Q and perturbation theory addresses the question of what such perturbations do to Q^{-1} which is the matrix from which V_1 and V_2 are obtained. A perturbation in Q may be represented by a perturbation in the identity matrix I multiplying Q since Q = IQ. The perturbed Q matrix is $Q_p = (I-P)Q$. The perturbations are associated with the identity matrix and given a negative sign so they are appropriate for the Neuman series. Perturbation theory is based on the Neuman series, which is an expansion of the inverse of a perturbed identity matrix I. P is a small perturbation in this matrix. The Neuman series:[1]

$$(I-P)^{-1} = I + P + P^2 + P^3 + \ldots.$$

For a small enough perturbation matrix P relative to I, the higher terms vanish relative to I. This concept is usually applied to rounding error which is very small (<0.1 percent) but applies well to the 5-percent tolerance we are using in our worst-case model. The implication from perturbation theory is that for small enough perturbations the inverse is linearly related to the perturbation. Namely:

$$(I-P)^{-1} = I + P.$$

The perturbed matrix $Q_p = (I-P)Q$ has an inverse $Q_p^{-1} = Q^{-1}(I-P)^{-1}$ from matrix theory. For a small enough perturbation, this becomes $Q_p^{-1} = Q^{-1}(I+P)$, which evaluates as $Q_p^{-1} = Q^{-1} + Q^{-1}P$. So, the change in the inverse of Q is linearly related to the inverse itself and is in fact just $Q^{-1}P$. This is the mathematical basis of our linearization.

Computational evidence indicates linearization is feasible for 5-percent component tolerance. The combined effect of multiple component tolerance may be modelled as the sum of the effect of each component tolerance individually. The 128 combinations of 5-percent extreme component tolerance are tested with this linearization technique to determine the worst-case combination. Although this linearized result is a reasonable approximation to the output voltages V_1 and V_2, a simple refinement is possible. This refinement is to calculate the exact network solution with the worst-case set of components. This method is much faster than solving the network problem 128 times.

---

[1] Stewart, G.W., Introduction to Matrix Computations (New York: Academic Press, 1973).

PROGRAM DESCRIPTIONS

DC WORST

This program generates the 5-percent tolerance worst-case plots. These plots outline the worst-case region. For each of the eight components the nominal fault line is generated by computing V_1 and V_2 for the range of 13 fault values from 0 to 50 ohms as shown in Table 2. The nonfaulty components are at their nominal values to produce the nominal fault line. For each of the 13 fault values it is desired to determine a combination of nonfaulty component tolerances which cause the point in V_1, V_2 space to move the furthest from the nominal fault line. Each nonfaulty component may be at either +5 percent or -5 percent tolerance. There are 2^7=128 combinations of tolerance conditions. The distance from the nominal fault line is examined for each of these combinations. The network is not solved exactly 128 times. A linearization technique based on the inverse of a perturbed identity matrix is used to speed this calculation process.[2] The origin shift and slope shift are precomputed for a +5 percent and -5 percent component tolerance for all the components. These shifts are accumulated for the combination of tolerances at hand. The program cycles through all 128 combinations until the one is found that shifts the measurement point the furthest from the nominal fault line. This combination is used to solve the network exactly. Summing a precalculated shift is far faster than solving the network. This is particularly significant since the calculation would have to be done 128 times to merely find the worst case. The worst-case points are plotted and connected to form the worst-case region.

DC CIRCUIT

This program is really the core of solving the Candidate DC Circuit. Given the driving currents and component values, this program determines the fault line slopes and voltages V_1 and V_2 at the output ports. It sets up the connection matrix, solves the circuit for the dependent variables by matrix inversion, and computes the desired results.

DC PLOT

This program initializes the plotting system and paints the background for most of the illustrations. The V_1 and V_2 axes are plotted and labelled and the fault line slopes labelled as G4 through G11.

ATN

This is a two-quadrant arctangent function not supplied by Hewlett-Packard. It returns an angle in degrees between +90 and -90, given the X and Y coordinate of a point.

---

[2] Stewart, G.W., Introduction to Matrix Computations (New York: Academic Press, 1973).

MATMULT

This simple subroutine multiples two real matrices. It simplifies matrix operations used in the program DC CIRCUIT.

MATINV

This subroutine uses a factor[3] approach to the Gauss-Jordan elimination algorithm to invert a matrix in place. Within the MATINV subroutine the subroutine LU generates the lower and upper triangular factors. The function FACTOR obtains permuted factors.

DISCUSSION OF 5-PERCENT FAULT REGIONS

Figures A-1 through A-16 in Appendix A show the fault regions for the 5-percent tolerance worst case for faulty components 4 through 11, for symmetric current sources, and then for asymmetric current sources.

There are several possibilities that these regions bring up when deciding if a given measurement belongs to one region or another. Especially near the origin when the faulty component is nearly nominal, all of the fault regions overlap. As the fault becomes more severe, the regions begin to separate and become distinct from each other. There are also forbidden regions where a measurement would not fall into any of the regions. Obtaining such a measurement implies a multiple component fault or component tolerance outside the 5-percent worst-case assumption. The type of decision criterion also makes a difference as to how these regions should be viewed. For example, a measurement point could fall directly inside a measurement region but using maximum likelihood it could be diagnosed to be in an altogether different fault region. This is possible because maximum likelihood normalized the distance measure by the standard deviation associated with a given fault.

DEFINITION OF THE 5-PERCENT TOLERANCE FAULT REGION

The 5-percent tolerance fault region is formed by fault incidents in which faulty components take on values from the list in Table 2. For each fault value the seven remaining nonfaulty components take on all 128 combinations of +5 percent or -5 percent tolerance and the computer simulation solves the dc circuit for port voltages V_1 and V_2. The combination which causes V_1 and V_2 to deviate the furthest from the fault line is chosen as the worst 5-percent tolerance case. The deviation is measured as Euclidean distance from the fault line with no component tolerance on the nonfaulty components. The method of this simulation is described in greater detail in Section 4, Worst-Case Simulation Method.

---

[3] Arden, Bruce W., and Kenneth N. Astill, Numerical Algorithms: Origins and Applications (Reading, Massachusetts: Addison-Wesley Publishing Company, 1970).

FAULT REGION FIGURES

The 5-percent worst-case tolerance fault regions shown in Figures A-1 through A-16 generally are finite two-dimensional trapezoids. In the special case of a component fault in R4 or R10, the trapezoid becomes infinitely long, as shown in Figures A-1, A-7, A-9 and A-15, respectively. These infinite trapezoids occur because R4 and R10 are series input resistors to the candidate circuit and are driven by constant current sources. As R4 or R10 ranges from a zero resistance (short) to an infinite resistance (open) the voltages V_1 and V_2 range from zero to infinity. The voltage drops across the remainder of the circuit, other than R4 and R10, do not change since the driving forces are two constant current sources. The fault region for component R11 is also a special case. When the driving current sources I_1 = 1 ampere and I_2 = 1 ampere are symmetric, the R11 fault region reduces to a bounded line in V_1, V_2, space as shown in Figure A-8. This line extends from V_1 = 0.05 volt and V_2 = 0.05 volt to V_1 = -0.05 volt and V_2 = -0.05 volt. When the driving current sources I_1 = 1 ampere and I_2 = 2 amperes are asymmetric, the R11 fault region assumes the trapezoidal form shown in Figure A-16. The apparent slight bowing out of the fault regions is an optical illusion related to Zollner's illusion.[4] Zollner's illusion is the earliest of the optical illusions and dates to 1860. It is manifest when parallel lines are hatched and no longer appear parallel to the eye.

ROBUSTNESS CONSIDERATION

The fault regions illustrated by Figures A-1 through A-16 may be used to gain some insight into the robustness of the Liu fault diagnosis approach. The regions illustrated are the extreme examples of the worst case of 5-percent component tolerance. All of the worst-case fault regions overlap each other, particularly near the origin. Single component faults R6, R8, and R11 residing in the interior of the circuit have little effect on the measured port voltages. In fact, with symmetric current sources, the value of R11 has no effect at all on the measured port voltages. Faults in R5, R7, or R9 have a substantial effect on the measured port voltages. With severe enough faults to these components, the faults can be diagnosed. The fault regions for R6 and R8 are very small and are bounded by less than 0.2 volt in V_1 and V_2. These fault regions overlap all of the others and would be difficult to diagnose. The fault region for component R11 is a bounded line with symmetrical current sources (Figure A-8). The fault region for component R11 with asymmetric current sources is a bounded trapezoid and is typical of the majority of the fault regions. In these worst-case examples, the Liu fault diagnosis technique does not appear to be robust in terms of immunity to component tolerance. A fault in R11 is completely undetectable if the current sources are symmetric. Even extreme faults in R6 or R8 fall within the same fault region as R4, R5, R7, R9, or R10. Faults R4, R5, R7, R9, and R10 are separable when at or near the extremes of the fault (open or short). These form a nonintersecting set when near the extremes. The robustness of other fault combinations may be estimated by comparing the fault regions involved. There are far too many possible combinations to treat in a detailed fashion. The saving grace of the Liu fault diagnosis is that these examples are not likely

[4] Tolansky, S., Optical Illusions (London: Permagon Press, 1964).

to occur. The probable case is described in the next subsection. The worst case is illustrated only as a severe challenge to the Liu diagnosis concept.

PROBABLE CASE

The 5-percent worst-case example for the candidate dc circuit assumes the worst possible combination of 5-percent component tolerance. Each component is at one 5-percent extreme or the other. It is interesting to compare this worst case with the probable case. The seven component distributions are assumed to be independent. The central limit theorem states:

> The sum of many independent random variables approaches a Gaussian-distributed variable.

Therefore, the sum of the effects of component tolerances on the port voltages V_1 and V_2 as contributed by the nonfaulty components tends toward a Gaussian distribution. This is also a consequence of the linear effect of small perturbations in the component values. The previous illustration and discussion have been for the case where all nonfaulty components take on their most deleterious 5-percent extremes. This is an extremely unlikely event in actual practice. The probable case includes the 68 percent of all component tolerance states falling within one standard deviation on an assumed joint Gaussian distribution.

In practice, +5 percent tolerance means that no component from a production run will be allowed to fall outside tolerance range.[5] If a component is outside this range it is by definition faulty. The manufacturing and sorting techniques employed for components tend to make the distribution of values bimodal. The values do tend to lie at the +5 percent or -5 percent extremes with few in the middle. If they should fall in the middle they would be sorted out for 2-percent or even 1-percent components. So the sorting technique actually removes most components not at the extremes, thereby leaving a bimodal distribution.

From the central limit theorem mentioned previously the effect of joint distribution of component tolerance on port voltages V_1 and V_2 tends towards a Gaussian with increasing numbers of components. Assuming a Gaussian distribution, we can outline the probable case of component tolerance to contrast with the worst case we have previously illustrated. Given a bimodal component distribution, the probability of +5 percent tolerance is 0.5 and the probability of -5 percent tolerance is 0.5. The joint probability that all seven nonfaulty components are at their worst-case is $(0.5)^7 = 0.0078$. From a table[6] of Integrals of the Gaussian normal error function, this is 2.42

---

[5] Westman, H.P., ed., Reference Data for Radio Engineers (New York: Howard W. Sams & Co., Inc., 1972).

[6] Holman, J.P., Experimental Methods for Engineers (New York: McGraw-Hill Book Company, 1978).

standard deviations from the mean. Since one standard deviation encloses 68 percent of all cases, the fault region represented by the probable case would shrink the region by a factor of 2.42 in both axes. The fault region area would shrink by a factor of 5.86. The overlap of the fault regions in this probable-case situation is very much less than the worst case we have illustrated. The likelihood of a correct fault diagnosis is much improved in the probable case.

SECTION 5

COMPARISON OF NEAREST NEIGHBOR RULE AND LIU APPROACH

FAULT DICTIONARY AND FAULT PATTERN APPROACHES

The benchmark for comparison with the Liu approach to fault diagnosis is the classical nearest neighbor rule. For the nearest neighbor rule a fault dictionary is constructed which consists of a set of preselected fault conditions. These faults must be discrete faulty component values. A short in some component may be one dictionary entry and an open in the same component may be another entry. Other less severe fault conditions in the same component may form further entries in the fault dictionary. To adequately cover all potential faults may require a very large dictionary. Associated with each dictionary fault is the set of measurements in multidimensional space that this fault would produce. The dictionary represents a set of points in the multidimentional measure space. Each measurement port may be measured at more than one frequency or may be complex and the number of dimensions may become very large. If the measurement is a complex voltage or current, the number of dimensions is doubled. The magnitude or phase alone of the measurement may be used, in which case the dimensions would not double.

Measurements are made on a test circuit to determine if one of the faults in the fault dictionary is present. The measurements on the test circuit constitute a point in the multidimensional space. The dictionary faults also reside as points in this space. The nearest neighbor rule associates the test point with the nearest (in the Euclidean sense) point in the fault dictionary set. This fault is diagnosed as the test circuit fault.

The Liu approach is quite different. In this approach a set of fault patterns is generated which corresponds to the fault dictionary of the nearest neighbor rule. These patterns are lines rather than points in multidimensional space. These lines are the locus of points formed in the measurement space when a component or branch takes in all possible values. For example, the fault pattern of a resistance is the line formed when it takes on values from zero through infinity. In a linear circuit the fault patterns are straight lines.

A test circuit measurement forms a point in the multidimensional measurement space. The fault line which is nearest (again in the Euclidean sense) to the point is the fault diagnosed for the circuit. For a circuit with all nominal values except for the fault, the measurement point will be exactly on one of the fault lines. Particularly when all components are nominal, the Liu approach can diagnose not only which component is faulty but what value it has taken on. This is because the locus of points bears a one-to-one relationship with the value of the faulty component. This may not necessarily be true for a nonlinear circuit.

SEPARABILITY CONSIDERATION

The issue of separability of the fault dictionary entries in the nearest neighbor rule and the fault patterns in the Liu approach are considerably different. The first consists of a set of points and the second consists of a set of lines passing through the origin. The origin is the point in measurement space for which all components are nominal. These points or lines may form clusters which are either near each other in space or similar to each other in slope. The further apart in space the fault dictionary entries are, for the nearest neighbor rule, the more likely a fault may be correctly diagnosed. The greater the angular separation for the Liu fault patterns, the more likely this approach is to correctly diagnose a fault.

ROBUSTNESS CONSIDERATION

The robustness of a fault diagnosing approach is a measure of its immunity to tolerance in the nonfaulted components. The nearest neighbor and Ruey-wen Liu approaches do very well when the nonfaulted components are at exactly their nominal values. The shortcoming of either approach becomes evident when component tolerance is introduced. This is because the fault dictionary of the nearest neighbor approach or the fault pattern of the Liu approach assumes that the nonfaulted components are exactly at their nominal values. The fault diagnosis is a form of pattern recognition. Component tolerance blurs the template which is being matched. In the nearest neighbor rule the dictionary points become clouds of points as the components take on a distribution of values. There is potential for overlap between these clouds. The fault patterns of the Liu approach become regions around the nominal fault line. Examples of this for the candidate dc circuit are illustrated by Figures A-1 through A-16 in Appendix A. These are for 5-percent tolerance worst cases for each of the fault patterns. For these worst cases the overlap of the fault regions is considerable, especially for soft faults where the fault is not extreme.

SECTION 6

CONCLUSIONS AND RECOMMENDATIONS

The canonical nearest neighbor rule and the fuzzy distance measure attributed to Dr. Samuel Bedrosian of the University of Pennsylvania are fault diagnosis approaches based on a fault dictionary. These approaches hypothesize a set of fault conditions in which a component or components take on a fault state with specific values. The nearest neighbor rule and Bedrosian fuzzy distance measure approaches both match the test circuit measurements to one or more of the hypothesized faults in the fault dictionary. The limitation is that a component may fail in a way other than the finite number of possibilities in the fault dictionary and not be properly diagnosed.

The Liu fault pattern approach has the tremendous advantage of representing an infinite and continuous number of fault conditions for a single component fault. Its disadvantage is the restriction to a single component fault situation. This is not considered a severe limitation since multiple faults are not common enough to be of great concern.

Immunity to component tolerance is not a simple issue because it is so dependent on the individual circuit topology and the individual components in that circuit. In the fault dictionary approaches, the faults are represented by points in the measurement space. Component tolerance on a statistical basis replaces these points by more or less spherical clouds centered on the points. The possibility exists that these clouds will intersect making diagnosis unreliable. In the Liu fault pattern approach, the faults are represented by lines which are the locus of points in measurement space as the fault goes through a range of values. Component tolerance on a statistical basis replaces those lines by more or less cylindrical clouds centered on the lines. These cylindrical clouds all pass through the measurement space origin. Near that origin all of the clouds overlap making diagnosis of soft faults unlikely. As the severity of the faults increase and we move away from the common origin, the cylinders become distinct from one another. Fault diagnosis may become reliable in this case. The Liu fault pattern approach is not fundamentally more immune to component tolerance than the fault dictionary approaches. However, the Liu fault pattern approach is superior because it can diagnose a continuum of fault conditions.

BIBLIOGRAPHY

BOOKS

Arden, Bruce W., and Kenneth N. Astill, Numerical Algorithms: Origins and Applications (Reading, Massachusetts: Addison-Wesley Publishing Company, 1970).

Burington, Richard Stevens, Handbook of Mathematical Tables and Formulas (New York: McGraw-Hill Book Company, 1965).

Close, Charles M., The Analysis of Linear Circuits (New York: Harcourt, Brace & World, Inc., 1966).

Feller, William, An Introduction to Probability Theory and its Applications Vol. 1 (New York: John Wiley & Sons, 1965).

Holman, J.P., Experimental Methods for Engineers (New York: McGraw-Hill Book Company, 1978).

Lewis, W.E., and D.G. Pryce, The Application of Matrix Theory to Electrical Engineering (London: E. & F.N. Spon, 1965).

O'Nan, Michael, Linear Algebra (New York: Harcourt Brace Jovanovich, Inc., 1976).

Rogers, F.E., Topology and Matrices in the Solution of Networks (London: Iliffe Books Ltd., 1965).

Schwartz, Mischa, Information Transmission, Modulation, and Noise (New York: McGraw-Hill Book Company, 1970).

Smith, Ralph J., Circuits, Devices, and Systems (New York: John Wiley & Sons, Inc., 1967).

Stewart, G.W., Introduction to Matrix Computations (New York: Academic Press, 1973).

Tolansky, S., Optical Illusions (London: Permagon Press, 1964).

Westman, H.P., ed., Reference Data for Radio Engineers (New York: Howard W. Sams & Co., Inc., 1972).

ARTICLES

Huang, Z.F., C. Lin, and R. Liu, "Node-Fault Diagnosis and a Design of Testability," Technical Report No. 811, Department of Electrical Engineering, University of Notre Dame, Notre Dame, Indiana (July 27, 1981).

Liu, Ruey-Wen, and C.S. Lin, "Fault Diagnosis on a Navy Candidate Circuit," University of Notre Dame, Notre Dame, Indiana (March 26, 1982).

APPENDIX A

WORST-CASE FAULT REGION PLOTS

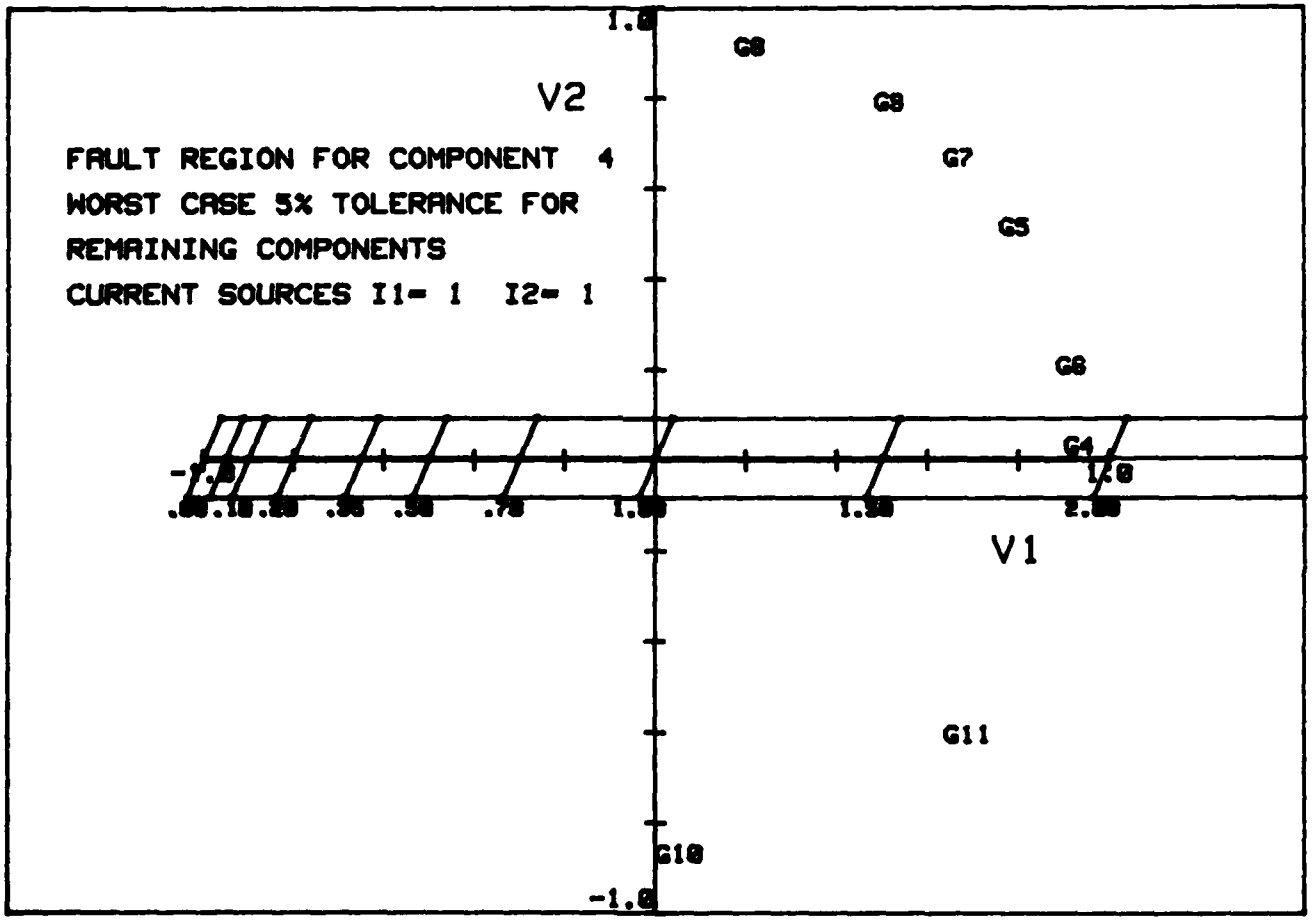


Figure A-1. Component 4 worst-case fault region (symmetric current sources).

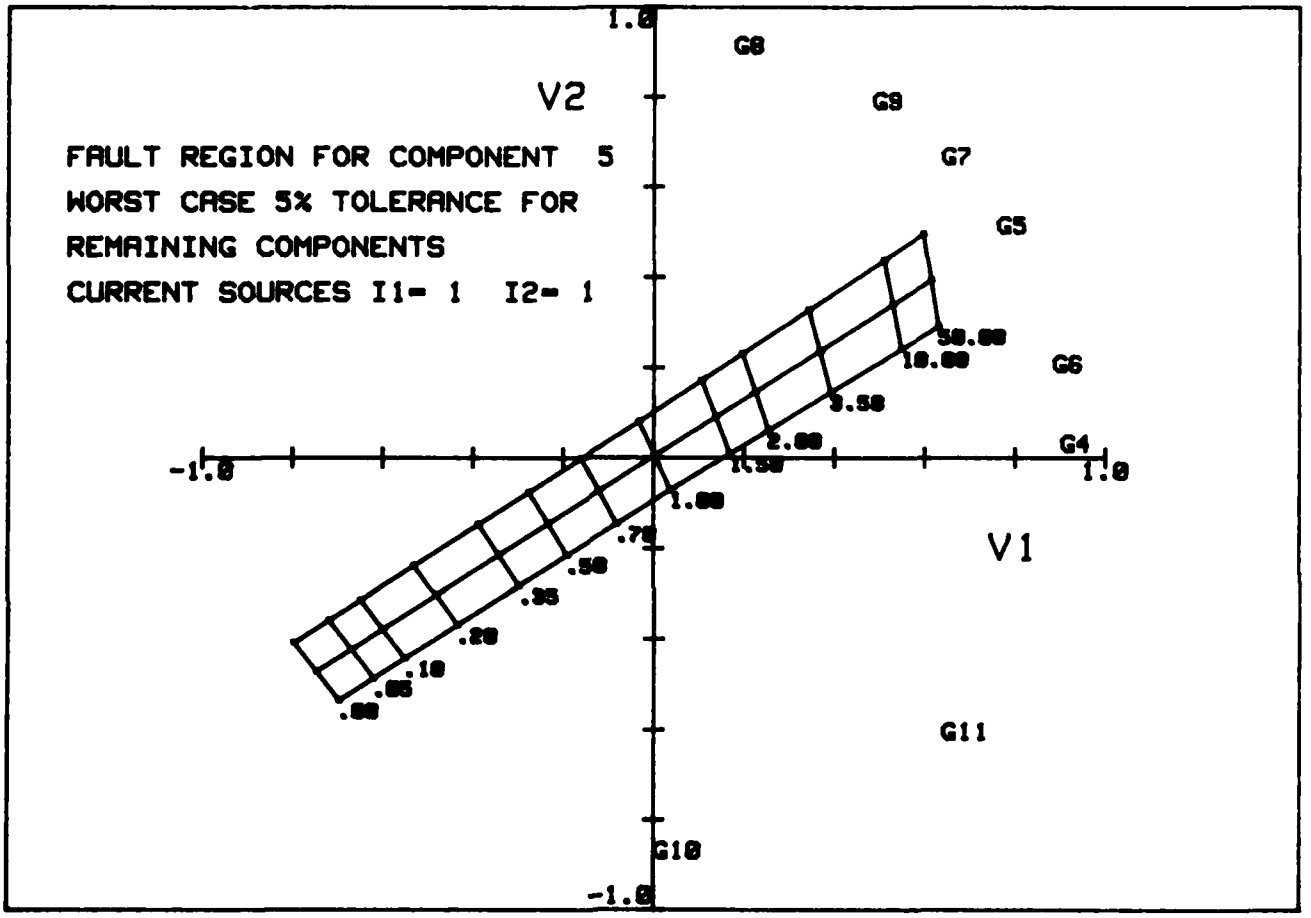


Figure A-2. Component 5 worst-case fault region (symmetric current sources).

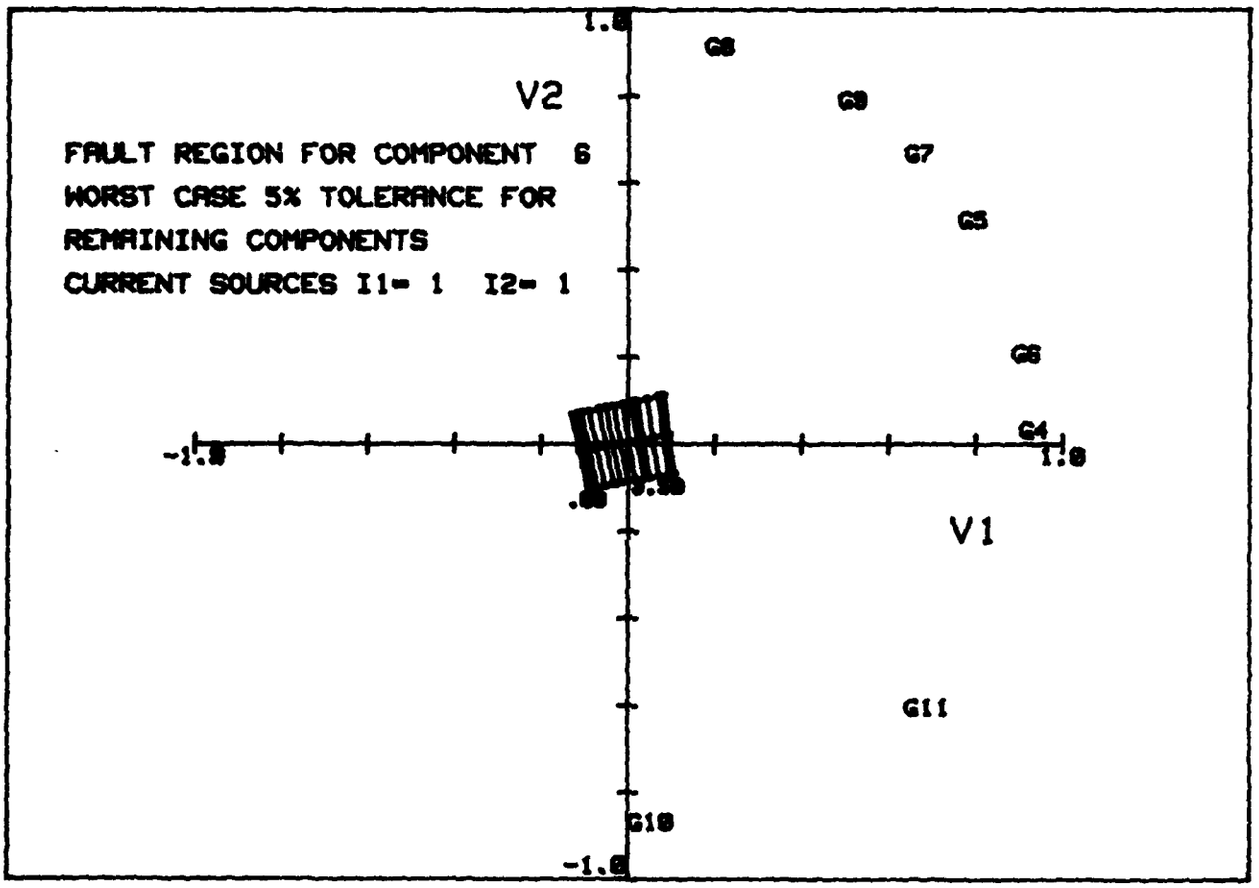


Figure A-3. Component 6 worst-case fault region (*symmetric* current sources).

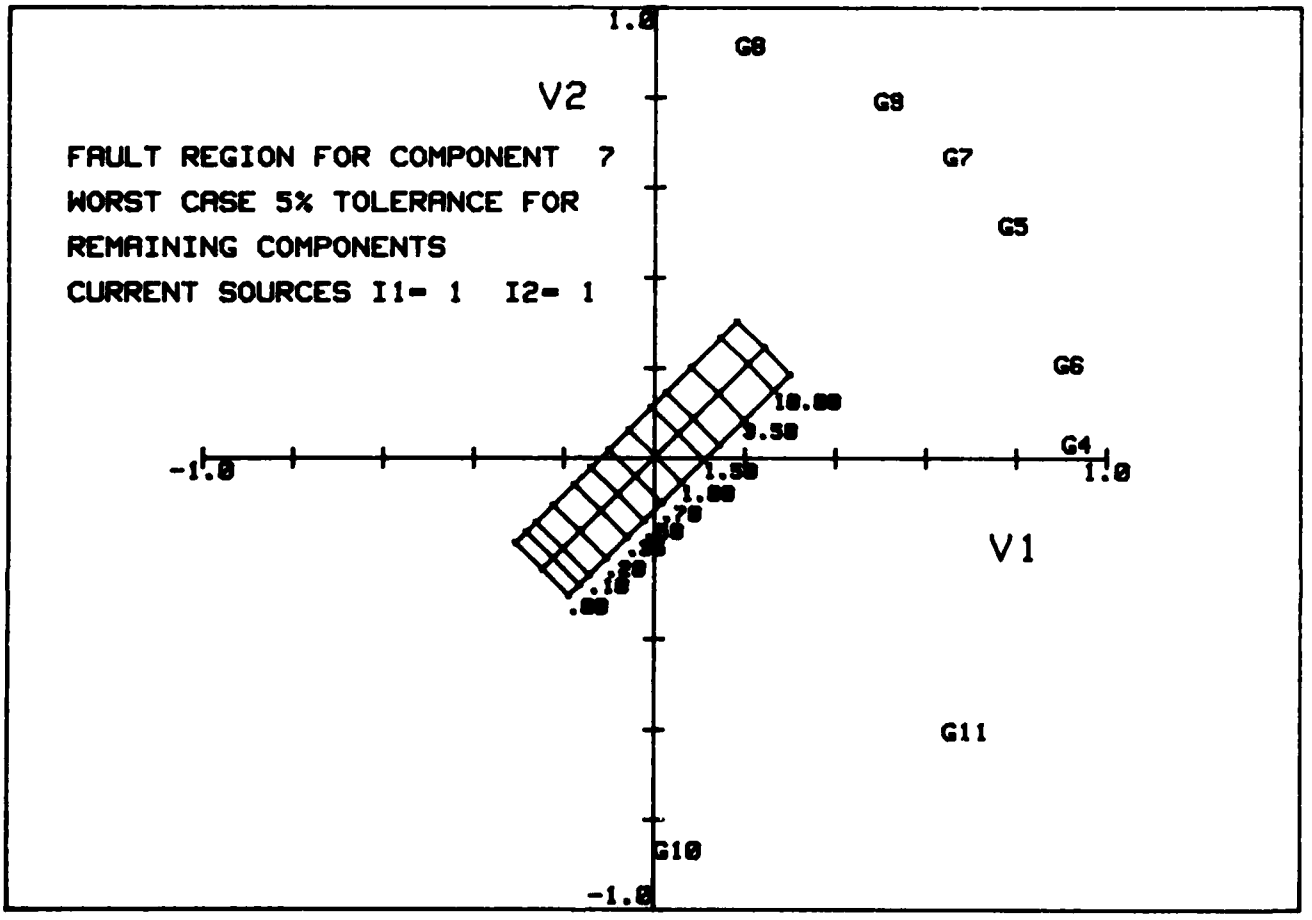


Figure A-4. Component 7 worst-case fault region (symmetric current sources).

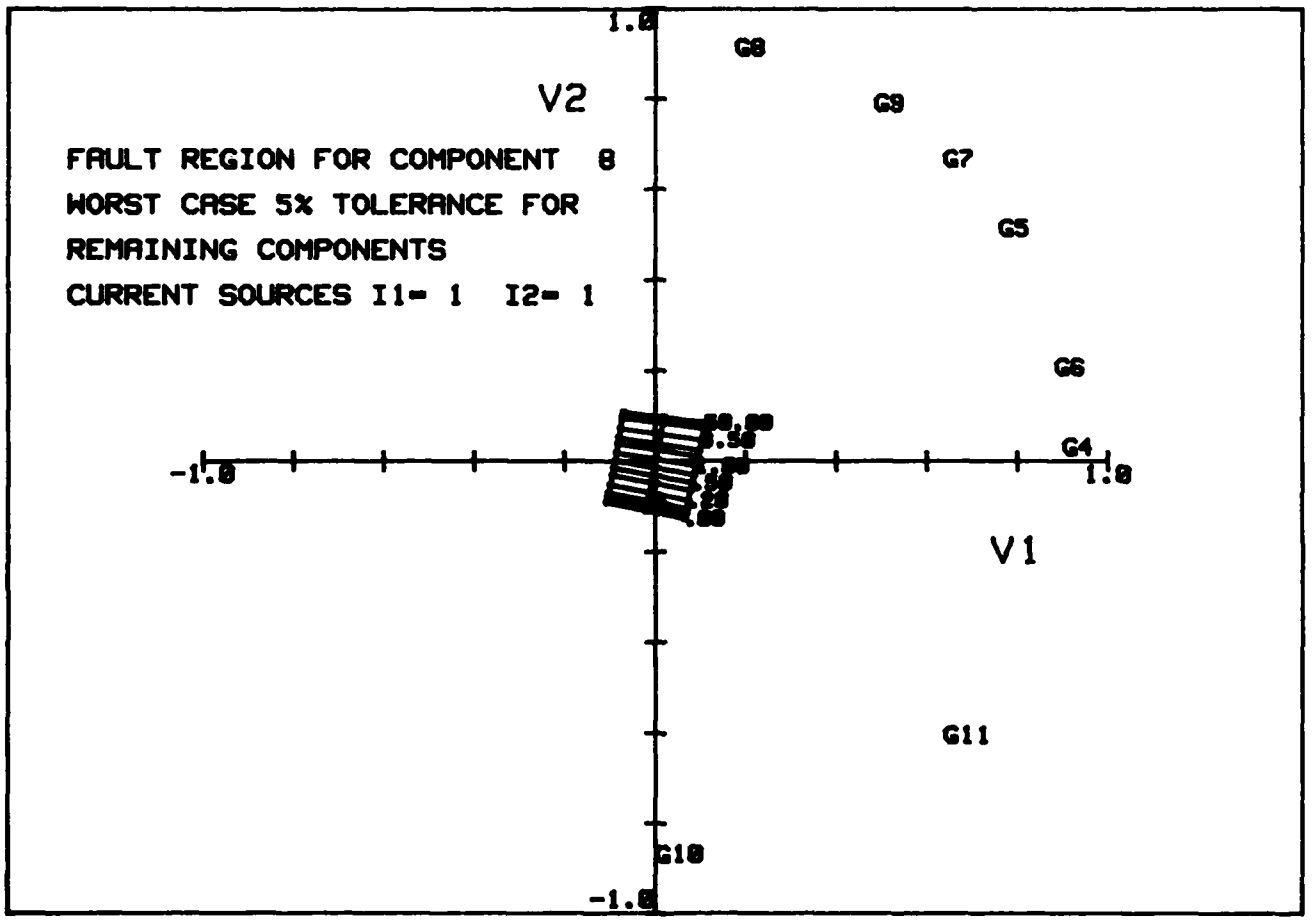


Figure A-5. Component 8 worst-case fault region (symmetric current sources).

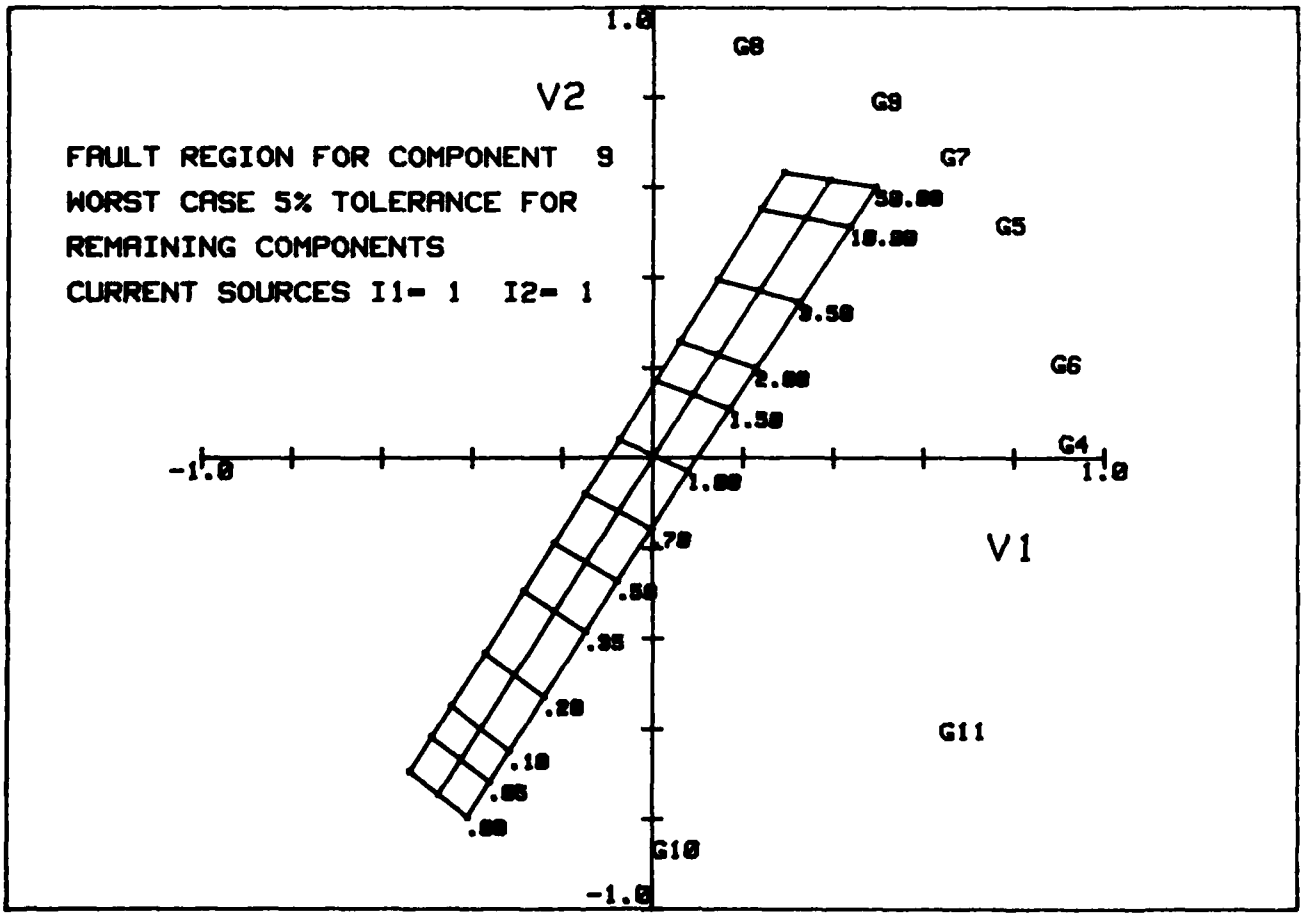


Figure A-6. Component 9 worst-case fault region (symmetric current sources).

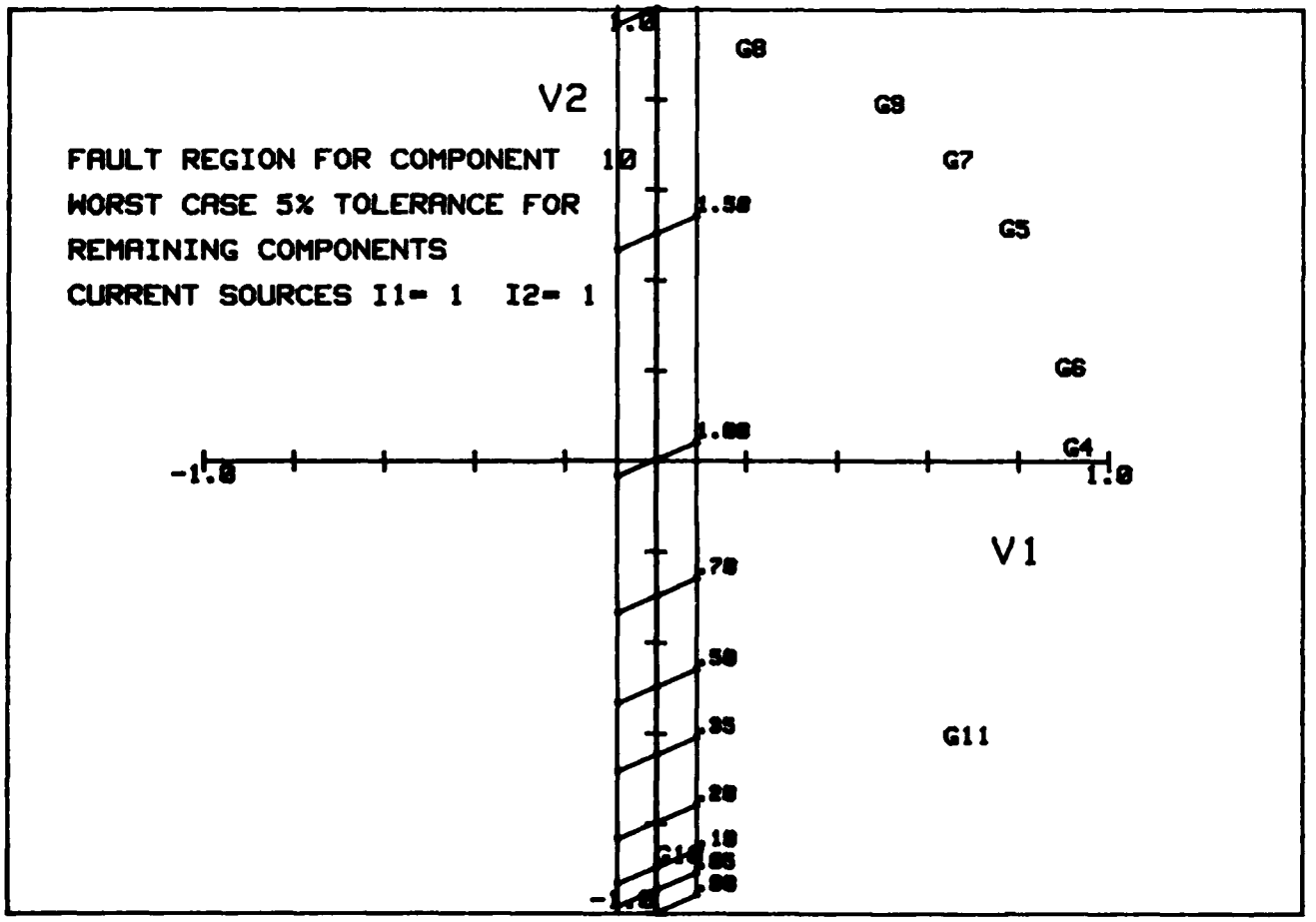


Figure A-7. Component 10 worst-case fault region (symmetric current sources).

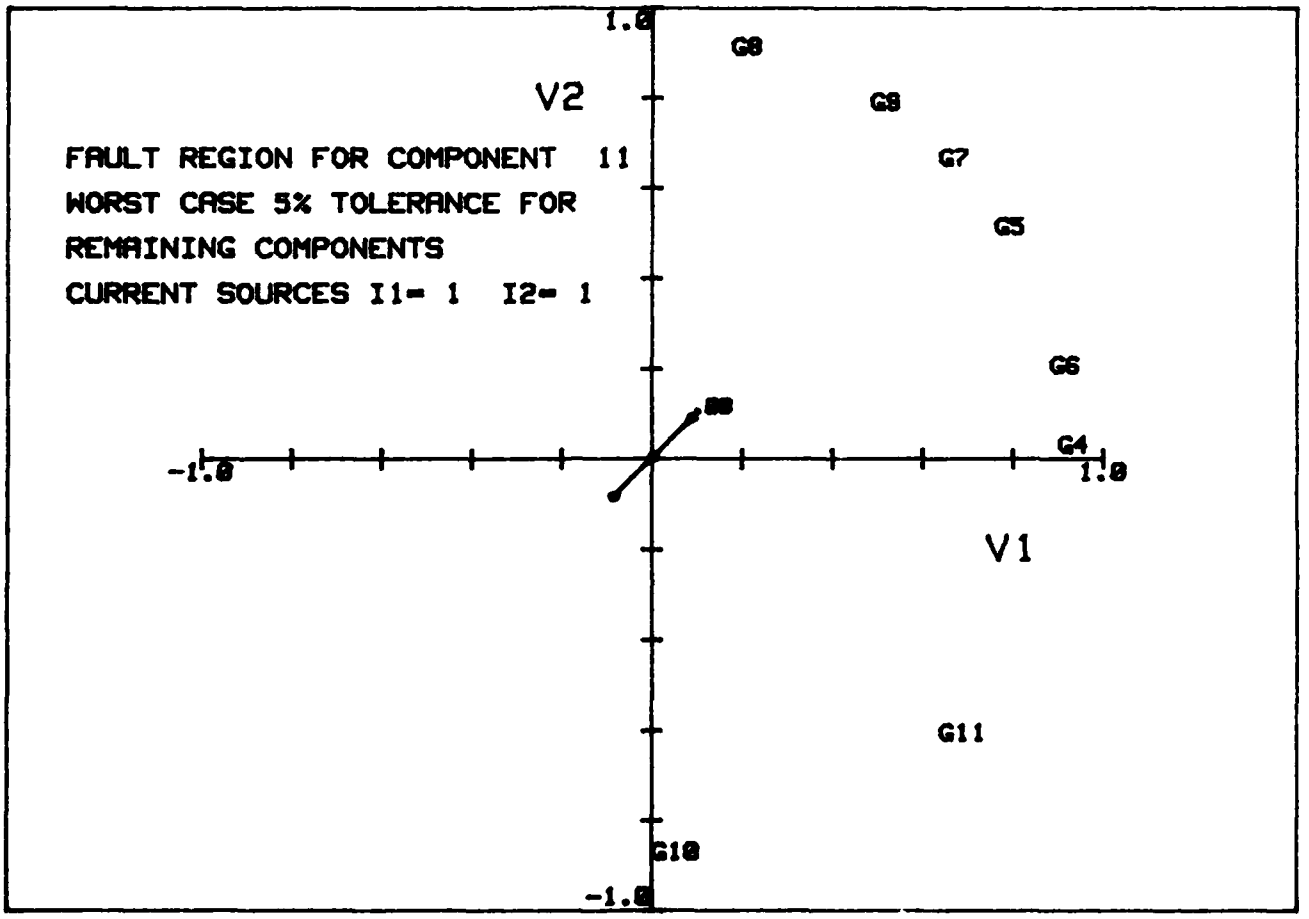


Figure A-8. Component 11 worst-case fault region (symmetric current sources).

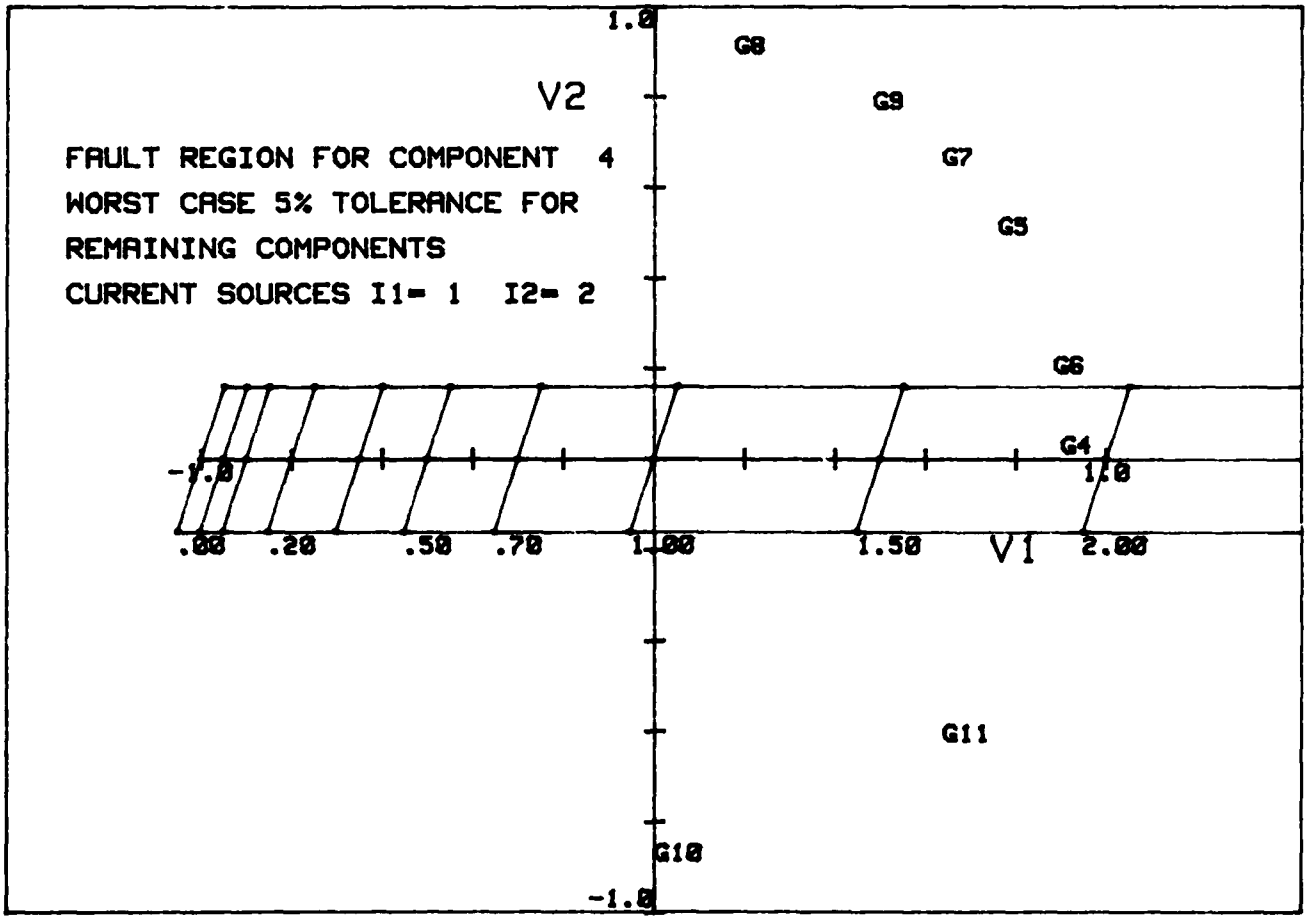


Figure A-9. Component 4 worst-case fault region (asymmetric current sources).

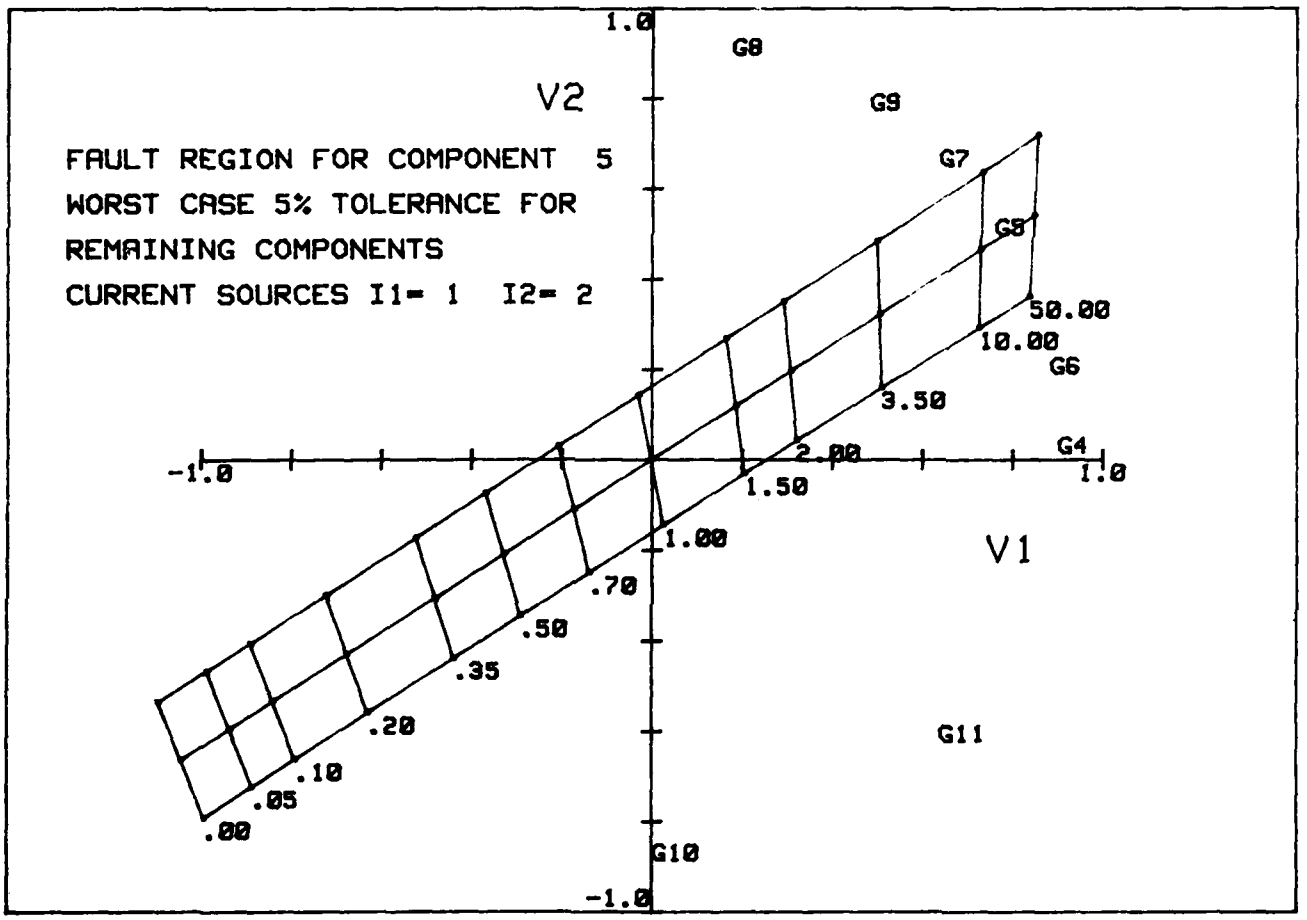


Figure A-10. Component 5 worst-case fault region (asymmetric current sources).

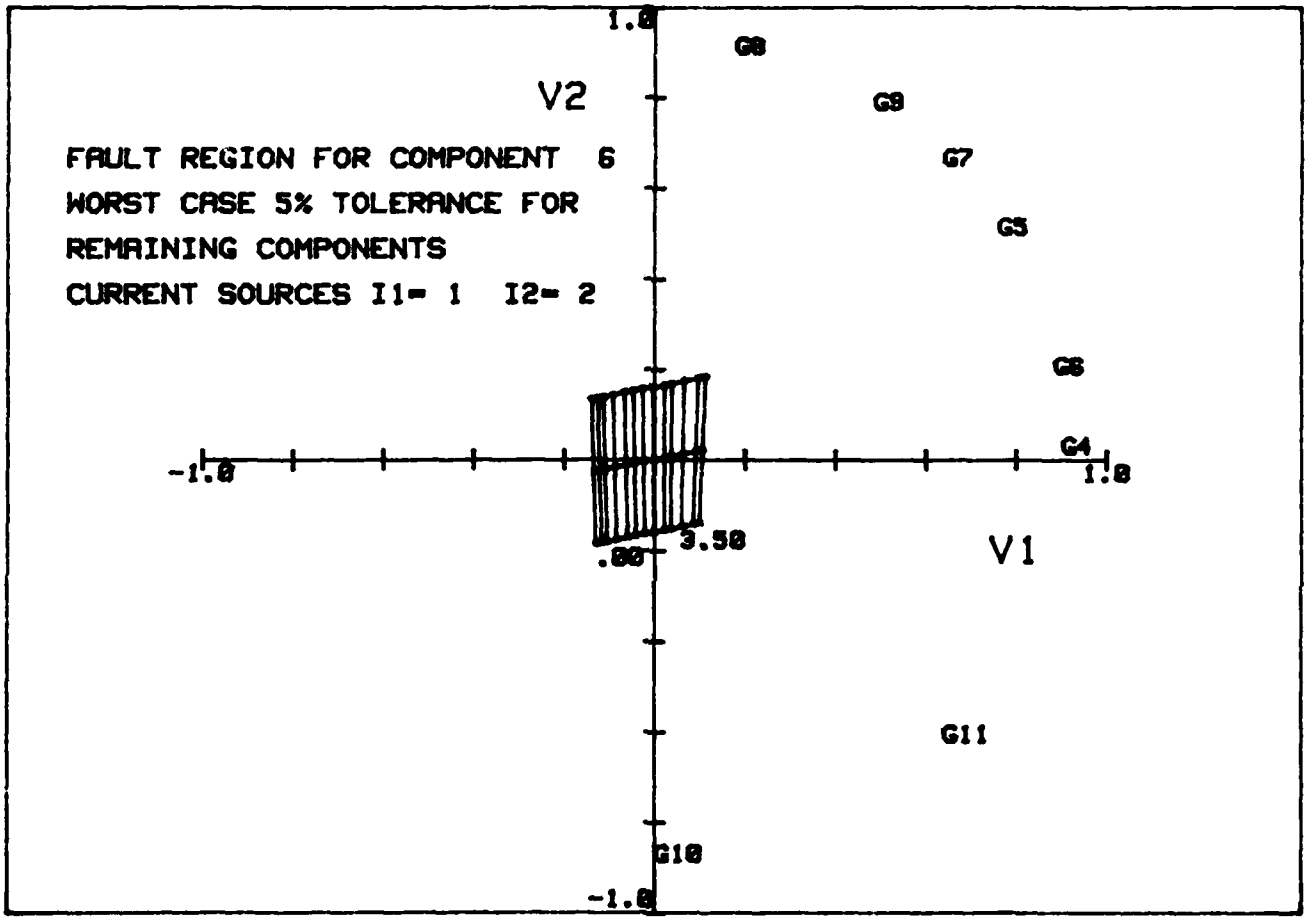


Figure A-11. Component 6 worst-case fault region (asymmetric current sources).

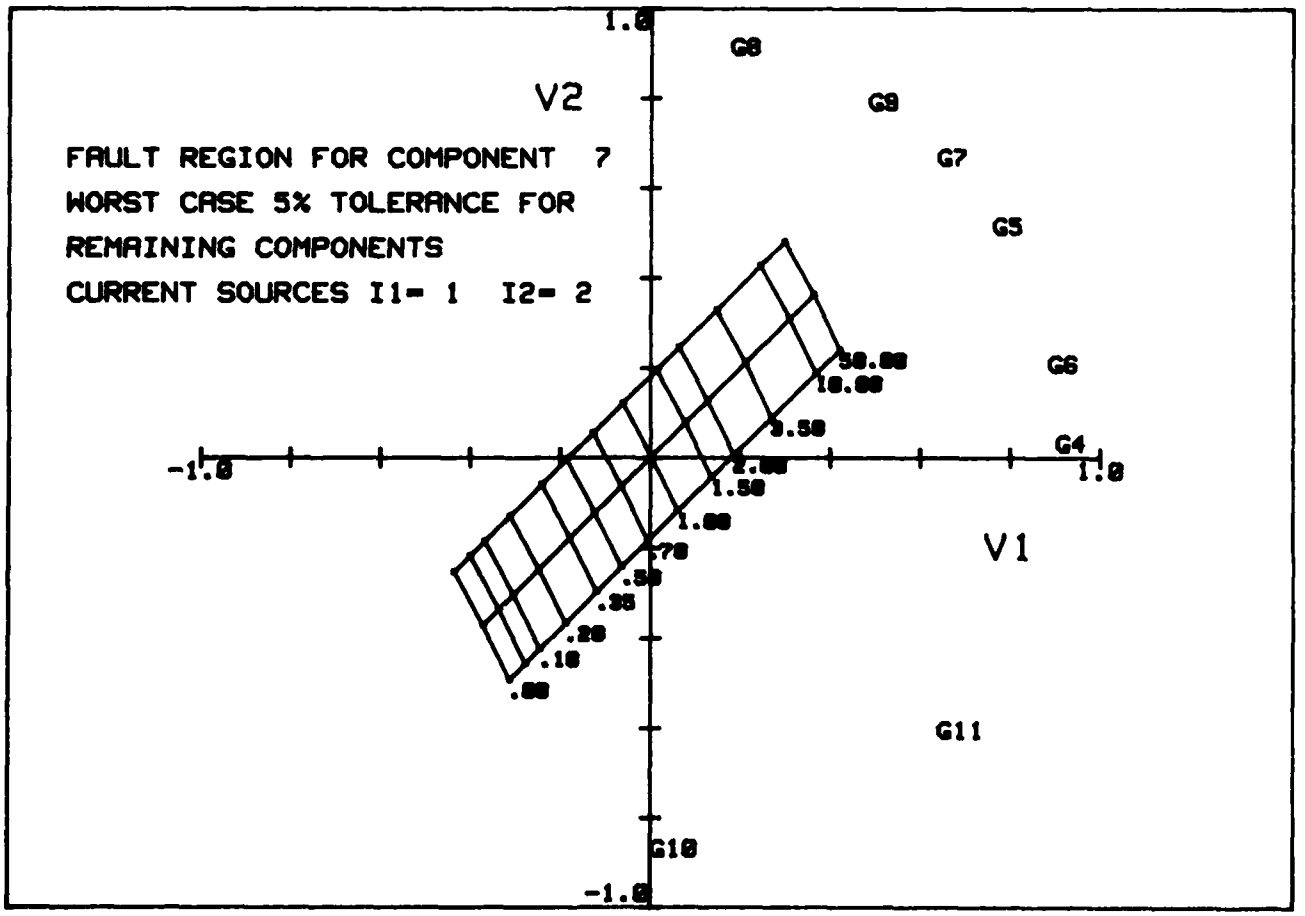


Figure A-12. Component 7 worst-case fault region (asymmetric current sources).

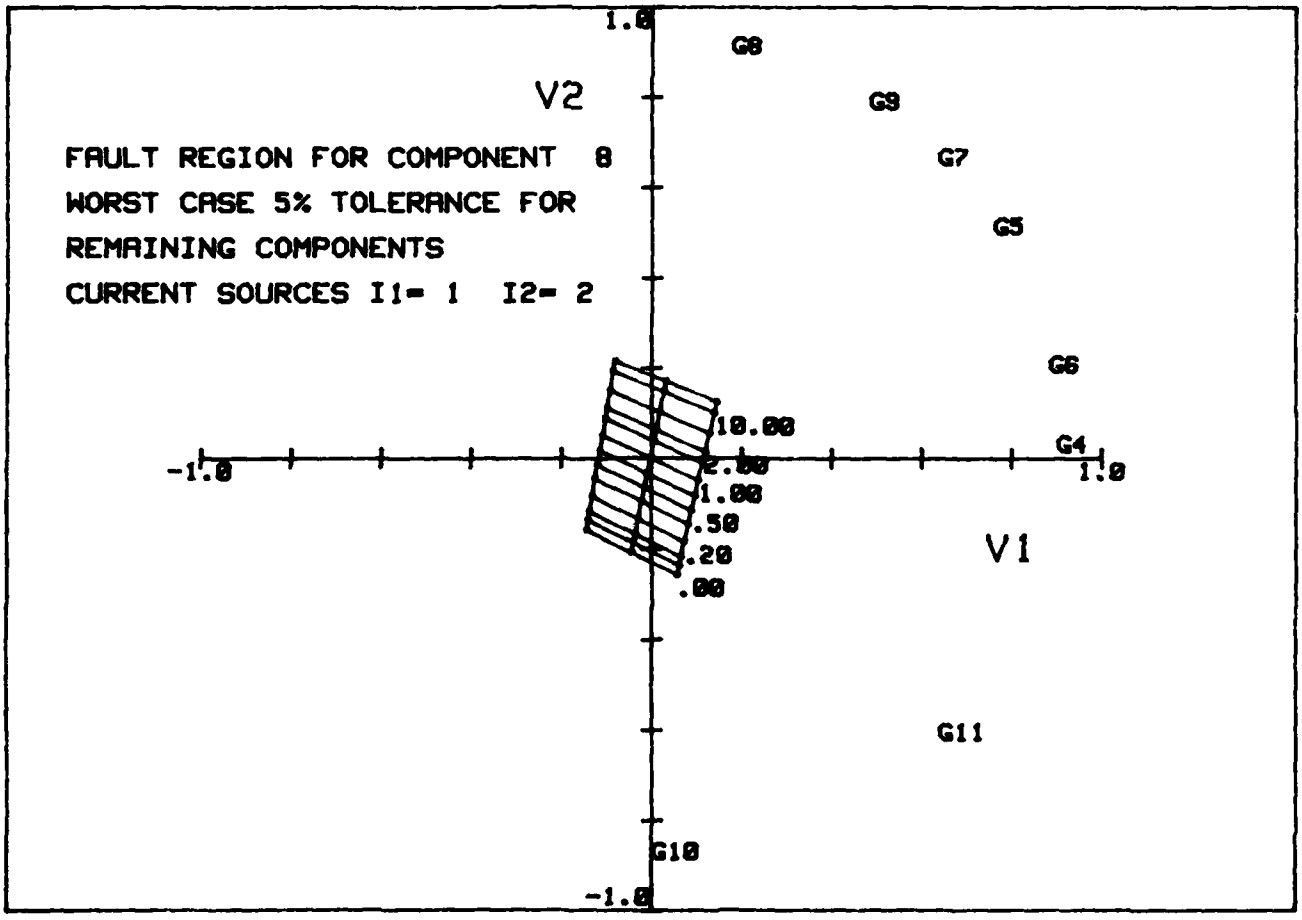


Figure A-13. Component 8 worst-case fault region (asymmetric current sources).

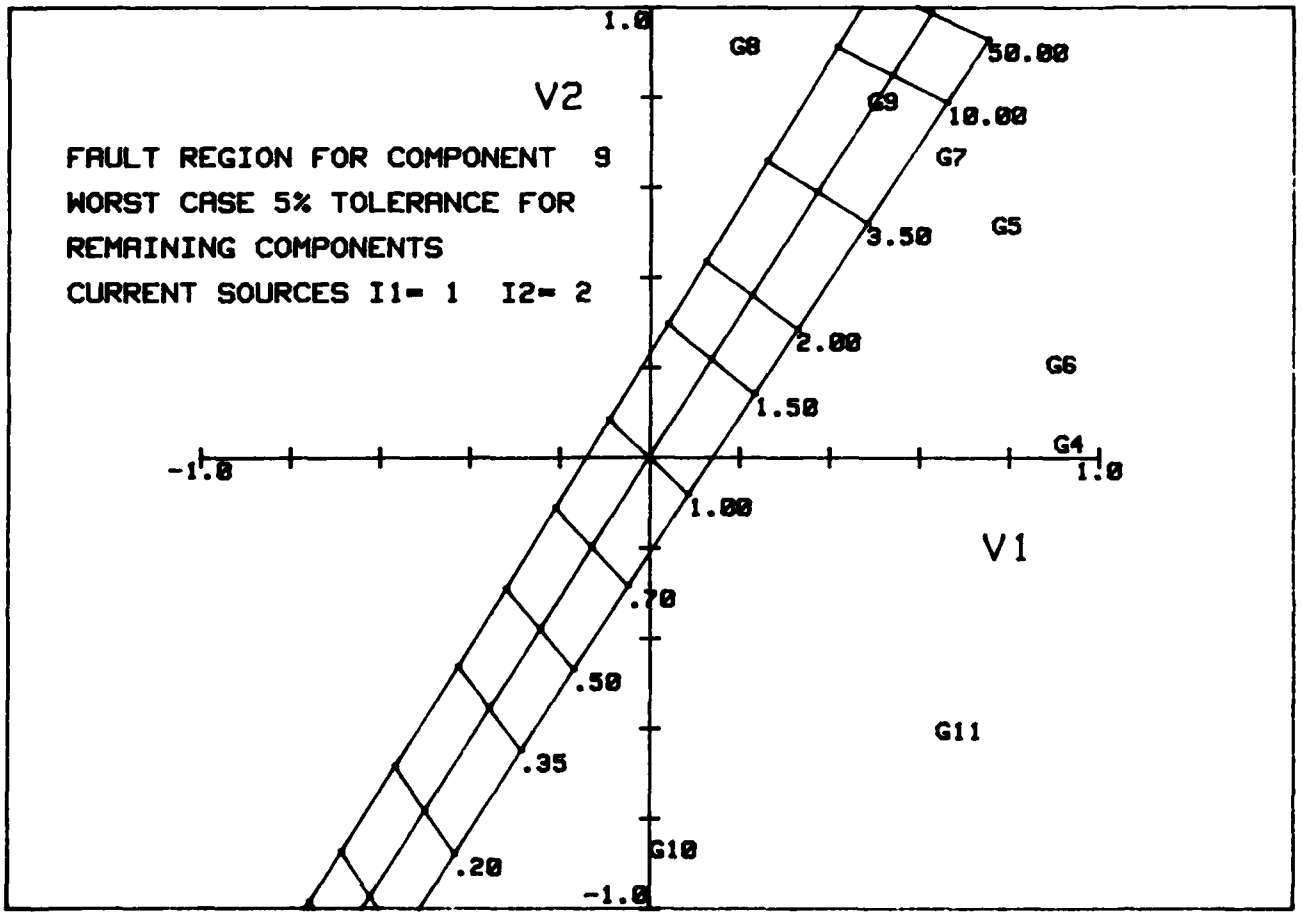


Figure A-14. Component 9 worst-case fault region (asymmetric current sources).

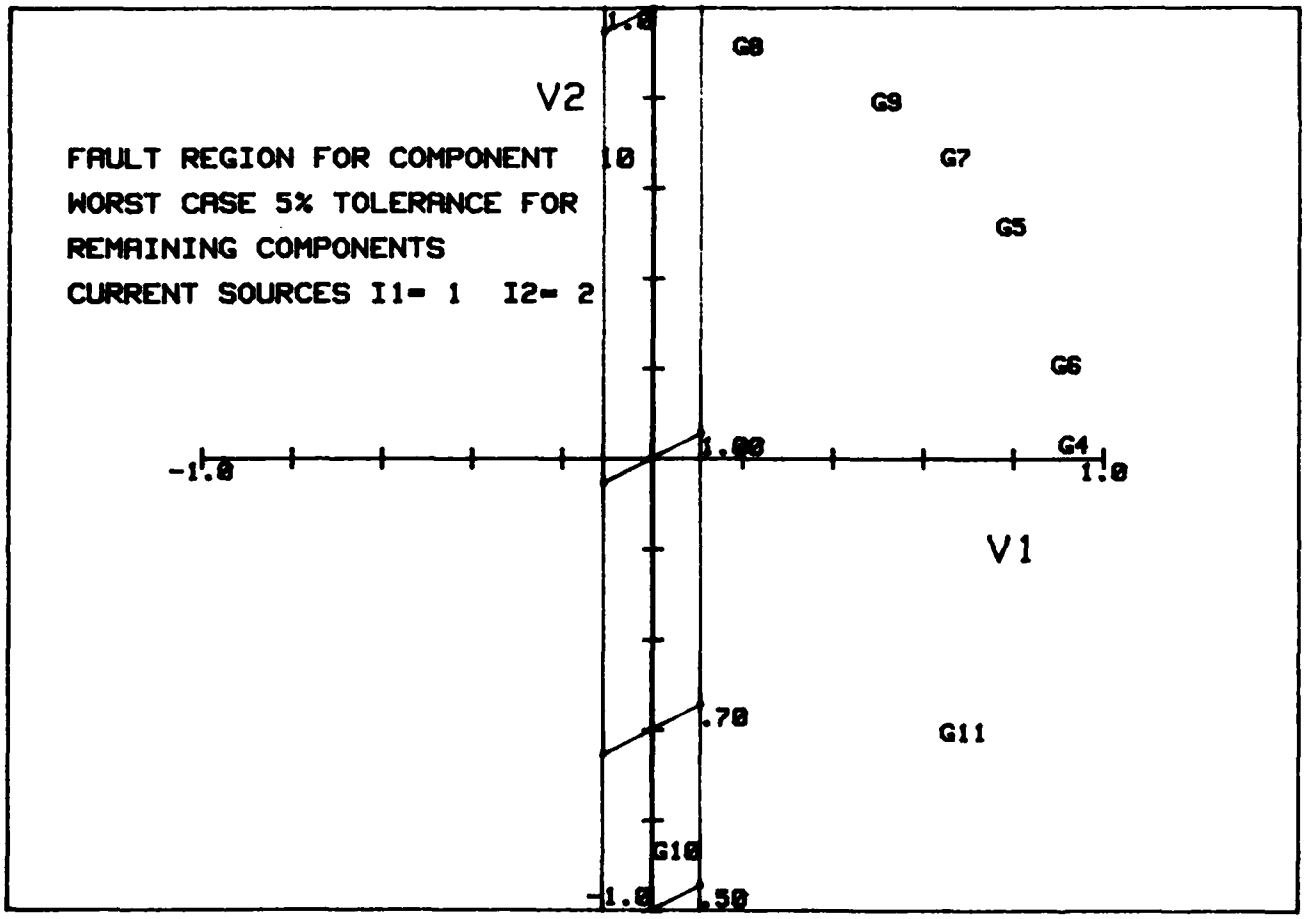


Figure A-15. Component 10 worst-case fault region (asymmetric current sources).

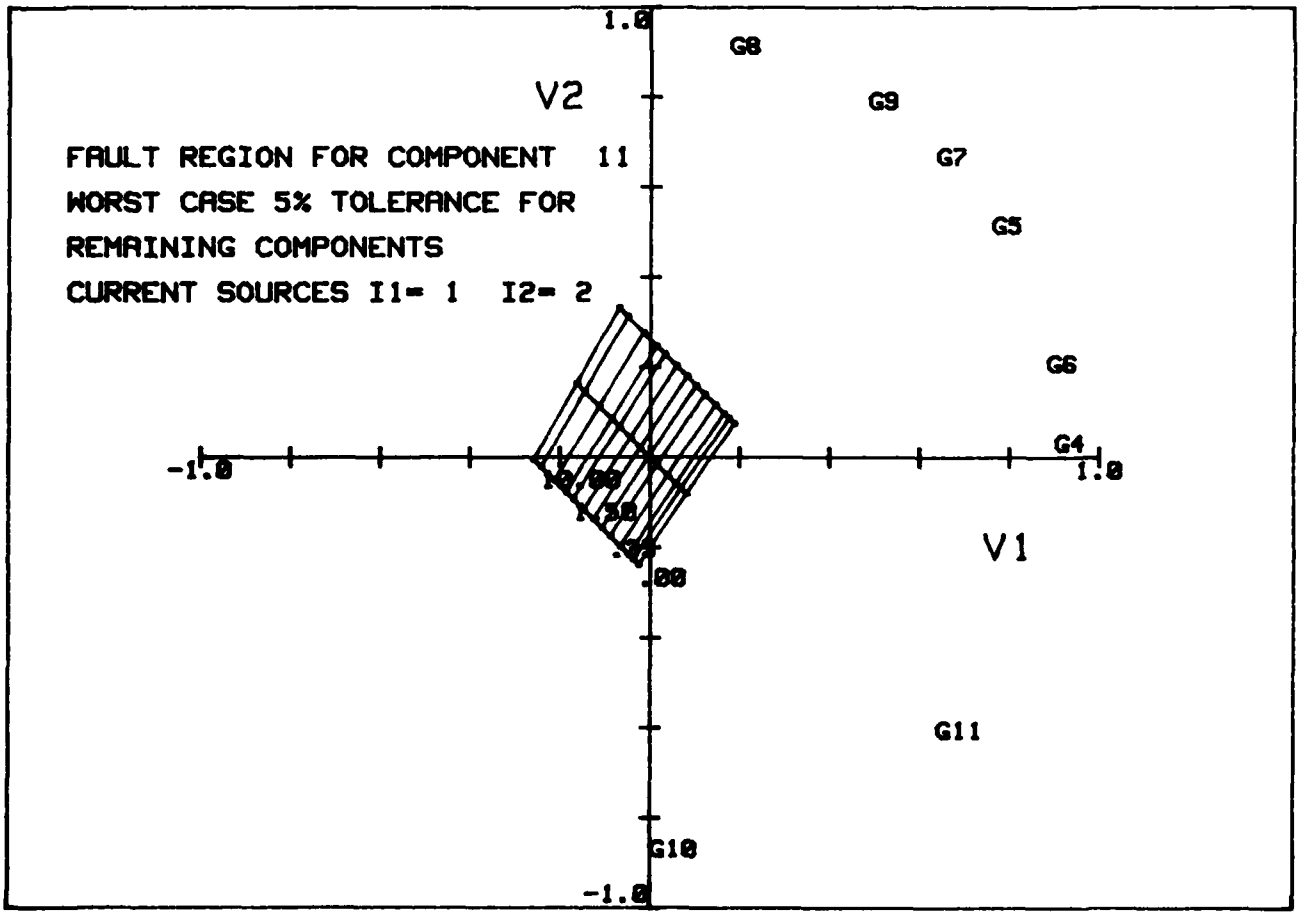


Figure A-16. Component 11 worst-case fault region (asymmetric current sources).

APPENDIX B

COMPUTER HARDWARE DESCRIPTION

APPENDIX B

COMPUTER HARDWARE DESCRIPTION

HP 9826 PROCESSOR

The HP 9826 Computer System is a high-performance instrument controller with built-in display and mass storage. It is a desktop system with a 178 mm diagonal CRT (cathode ray tube) display and 5.25-inch, 264 K-byte flexible mini disk drive.

The system is based on the 16-bit 68000 state-of-the-art microprocessor chip. The clock rate is 8 MHz. The 68000 has 32-bit internal architecture and 16-bit external architecture. There are 65 K-bytes of high-speed RAM (random access memory) in this configuration of the HP 9826.

The keyboard has 10 (20 with shift) user-defined softkeys, an HP-IB interface, a real-time clock, a programmable beeper, and a 400 x 300 pixel graphics display system. An unusual feature of the HP 9826 is the "knob" which provides the user with an input device for editing or inputting data which has an analog feel about it.

HP 9826 ENHANCED BASIC 2.0 LANGUAGE SYSTEM

The BASIC 2.0 language adds to the inherent simplicity of BASIC the computational power found in FORTRAN, ALGOL, and APL. High-performance I/O intensive constructs give this enhanced BASIC the highest performance found in interactive instrument control systems. This enhanced BASIC language is quite user friendly and provides immediate user feedback on a line-by-line basis. Erroneous syntax is detected and reported as each line is entered. Erroneous lines may not even be entered into a program, as they are caught at once. Variable names may be up to 15 characters, which is useful for self-documenting programs. Statements are numbered but may also carry a descriptive statement label. This too is useful for self-documentation of enhanced BASIC programs. Local subroutines and function subprograms may be included. This is a very powerful enhancement.

Additional enhancements include structured programming constructs, labelled COM (common) statements, and extensive debug and trace tools. Programs may be single-stepped or traced with a line-by-line log of line numbers and variable changes while the program is running.

HP 2671G THERMAL PRINTER

The Hewlett-Packard HP 2671G is a companion printer to the HP 9826 desktop computer. Characters or graphics are formed with a printhead containing 15 thin-film resistors arranged in a vertical column. Characters are formed on a high-resolution 18 x 15 dot matrix. Raster graphics resolution is 90 dots per inch both vertically and horizontally. Up to 720 dots are available across the page. A throughput of 16,200 dots/second is made possible by microcomputer control and buffering of the print mechanism.

An 8-bit HP-IB parallel interface is the link to the HP 9826 computer. The microcomputer accepts and buffers characters, commands, and graphics data over this interface. Handshake lines are used for intercept and acknowledgment of data transfers.

The HP 2671G has many printing modes and features. The normal print mode is 10 characters/inch (80 columns per line), which is printed at 120 characters/second. In the expanded mode there are 5 characters/inch (40 columns per line). In the compressed mode characters may be printed at 16.2 characters/inch (132 columns per line). Print enhancements include underlining, framing, and triple-pass bolding. Combinations of these print styles can even be intermixed on a single line. In the character mode, printing is bidirectional for greater speed. Character sets are available in foreign languages. Page formatting features include margins, tabs, and vertical page formatting. Graphics may be made directly from the HP 9826 graphics screen and may be autocentered, windowed, offsetted, and expanded.

HP 9872B PLOTTER

The HP 9872B plotter is a four-pen microprocessor-controlled flat bed plotter designed to create professional quality graphics for business and industry. It is driven by the HP 9826 processor with BASIC commands through an HP-IB interface. The graphics commands sent on the interface are in the Hewlett-Packard Graphics Language (HP-GL) which offers easy-to-remember mnemonic instructions. The BASIC commands are at a high level and actually generate the HP-GL instructions for the user.

Under program control the HP 9872B plotter can fetch any of its four pens and return the pens to their holders. This can also be done under manual control. Programmable pen velocity control is also featured. The extraordinary quality of graphics is a result of precise pen control movement through the use of sophisticated electronic circuitry. The addressable resolution of 0.25 mm (0.001 inch) and the precise repeatability of the HP 9872B insure superb quality plots on both paper and transparency film media up to 297 x 420 mm (11 x 17 inches). Pen selection and capping are fully under program control, allowing multicolor plotting. The plotter is user friendly and convenient.

APPENDIX C

PROGRAM LISTINGS

```
10     ! WORST CASE ANALYSIS PROGRAM
20     ! DC_WORST
30     !
40     !
50     ! THIS PROGRAM PRODUCES 5% WORST CASE PLOTS
60     ! AND PERFORMS THE CIRCUIT SOLUTIONS
70     ! AND WORST CASE ANALYSIS NEEDED FOR THESE
80     ! PLOTS
90     !
100    !
110    !
120    ! THE DATA STATEMENTS FOR ORIGIN_LOW,
130    ! ORIGIN_HIGH, ANGLE_LOW, AND ANGLE_HIGH
140    ! ARE RE-CALCULATED AND OVERWRITTEN
150    ! BY THE PROGRAM.
160    !
170    ! THESE DATA STATEMENTS WERE USED IN
180    ! PROGRAM DEVELOPMENT AND HAVE BEEN
190    ! RETAINED FOR INFORMATIONAL PURPOSES.
200    !
210    !
220    DIM Origin_low(4:11,1:2) ! ORIGIN SHIFT
230    !          FOR -5% COMPONENT TOLERANCE
240    !
250    DATA -.0500, .0000,  -.0177,-.0112
260    DATA -.0027,-.0005,  -.0064,-.0064
270    DATA -.0005,-.0027,  -.0112,-.0177
280    DATA  .0000,-.0500,   .0000, .0000
290    !
300    READ Origin_low(*)
310    !
320    DIM Origin_high(4:11,1:2) ! ORIGIN SHIFT FOR +5% COMPONE
NT TOLERANCE
330    !
340    DATA  .0500, .0000,   .0167, .0106,
350    DATA  .0025, .0005,   .0061, .0061,
360    DATA  .0005, .0025,   .0106, .0167,
370    DATA  .0000, .0500,   .0000, .0000
380    !
390    READ Origin_high(*)
400    !
410    DIM Angle_low(4:11,4:11)  ! VECTOR ANGLE FOR          -5%
 TOLERANCE ON COMPONENTS
420    !
430    DATA 00.00, 32.47, 11.31, 45.00, 78.69, 57.53,-90.00,-45
.00  ! R4
440    DATA 00.00, 32.47, 10.70, 45.25, 78.92, 57.9, -90.00,-46
.11  ! R5
450    DATA 00.00, 32.56, 11.31, 44.76, 79.49, 57.56,-90.00,-45
.37  ! R6
```

```
460    DATA 00.00, 32.37, 11.98, 45.00, 78.02, 57.63,-90.00,-45
.00   ! R7
470    DATA 00.00, 32.44, 10.51, 45.24, 78.69, 57.44,-90.00,-44
.63   ! R8
480    DATA 00.00, 32.10, 11.08, 44.75, 79.27, 57.53,-90.00,-43
.89   ! R9
490    DATA 00.00, 32.47, 11.31, 45.00, 78.69, 57.53,-90.00,-45
.00   ! R10
500    DATA 00.00, 32.88, 12.22, 45.00, 77.78, 57.12,-90.00,-45
.00   ! R11
510    !
520    READ Angle_low(*)
530    !
540    DIM Angle_high(4:11,4:11)  ! VECTOR ANGLE FOR +5% COMPON
ENT TOLERANCE
550    !
560    DATA 00.00, 32.47, 11.31, 45.00, 78.69, 57.53,-90.00,-45
.00   ! R4
570    DATA 00.00, 32.47, 11.83, 44.77, 78.48, 57.19,-90.00,-43
.96   ! R5
580    DATA 00.00, 32.39, 11.31, 45.23, 77.94, 57.50,-90.00,-44
.65   ! R6
590    DATA 00.00, 32.57, 10.65, 45.00, 79.35, 57.43,-90.00,-45
.00   ! R7
600    DATA 00.00, 32.50, 12.06, 44.77, 78.69, 57.61,-90.00,-45
.35   ! R8
610    DATA 00.00, 32.81, 11.52, 45.23, 78.17, 57.53,-90.00,-46
.04   ! R9
620    DATA 00.00, 32.47, 11.31, 45.00, 78.69, 57.53,-90.00,-45
.00   ! R10
630    DATA 00.00, 32.07, 10.45, 45.00, 79.55, 57.93,-90.00,-45
.00   ! R11
640    !
650    READ Angle_high(*)
660    !
670    DIM Angle(4:11) ! nominal vector angleS
680    DATA 00.00, 32.47, 11.31, 45.00, 78.69, 57.53,-90.00,-45
.00
690    READ Angle(*)
700    !
710    DIM X(0:255),Y(0:255),Error_ang(0:255)
720    DIM Measure_dist(0:255)
730    DIM Delta_v(1:2,1:10)
740    DIM Xplot(1:16,1:3),Yplot(1:16,1:3)
750    !
760    DIM R_value(1:13)
770    DATA 0,.05,.10,.2,.35,.5,.7,1,1.5,2,3.5,10,50
780    READ R_value(*)
790    !
800    DIM R(4:11)  ! RESISTOR ARRAY
810 R:     DATA 1,1,1,1,1,1,1,.5
```

```
820         READ R(*)
830     !
840     !
850     COM /Worst/ Flag
860     IF Flag=0 THEN LOADSUB ALL FROM "MATMULT"
870     IF Flag=0 THEN LOADSUB ALL FROM "MATINV"
880     IF Flag=0 THEN LOADSUB ALL FROM "ATN"
890     IF Flag=0 THEN LOADSUB ALL FROM "DC_CIRCUIT"
900     IF Flag=0 THEN LOADSUB ALL FROM "DC_PLOT"
910     Flag=1
920     !
930         DEG  ! SET  DEGREE MODE FOR TRIG FUNCTIONS
940     !
950     RESTORE R
960     READ R(*)
970     I1=1
980     I2=1
990     !
1000    ! DETERMINE NOMINAL ORIGIN AND NOMINAL VECTOR ANGLES
1010    !
1020    CALL Dc_circuit(I1,I2,R(*),V1,V2,Delta_v(*))
1030    Origin(1)=V1
1040    Origin(2)=V2
1050    FOR Component=4 TO 11
1060        Angle(Component)=FNAtn(Delta_v(1,Component-3),Delta_
v(2,Component-3))
1070    NEXT Component
1080    !
1090    ! CALCULATE ORIGIN SHIFT AND VECTOR ANGLE
1100    ! FOR +5% AND -5% COMPONENT TOLERANCE
1110    ! CHANGING ONE COMPONENT AT A TIME
1120    !
1130    ! THIS CUSTOM CALCULATION REQUIRES 100 SECONDS ON THE HP
9826
1140    !
1150    FOR Component=4 TO 11
1160        RESTORE R
1170        READ R(*)
1180        R(Component)=.95*R(Component)    ! -5% COMPONENT TOLE
RANCE INTRODUCED
1190        CALL Dc_circuit(I1,I2,R(*),V1,V2,Delta_v(*))
1200             FOR Comp=4 TO 11
1210              Angle_low(Component,Comp)=FNAtn(Delta_v(1,Comp
-3),Delta_v(2,Comp-3))
1220             NEXT Comp
1230        Origin_low(Component,1)=V1-Origin(1)
1240        Origin_low(Component,2)=V2-Origin(2)
1250        RESTORE R
1260        READ R(*)
1270        R(Component)=1.05*R(Component)  ! +5% COMPONENT TOLE
RANCE
```

```
1280       CALL Dc_circuit(I1,I2,R(*),V1,V2,Delta_v(*))
1290            FOR Comp=4 TO 11
1300             Angle_high(Component,Comp)=FNAtn(Delta_v(1,Com
p-3),Delta_v(2,Comp-3))
1310            NEXT Comp
1320         Origin_high(Component,1)=V1-Origin(1)
1330         Origin_high(Component,2)=V2-Origin(2)
1340   NEXT Component
1350   !
1360   !
1370   !
1380   FOR Worst=8 TO 8
1390   BEEP
1400   WAIT 1.0
1410   BEEP
1420   OUTPUT 1;"POSITION PLOTTER PAPER - PRESS 'CONTINUE' WHEN
 READY"
1430   PAUSE
1440   CALL Dc_plot(Angle(*))
1450   CSIZE 3.5
1460   LORG 3
1470   MOVE -65,35
1480   LABEL "FAULT REGION FOR COMPONENT ";Worst
1490   MOVE -65,30
1500   LABEL "WORST CASE 5% TOLERANCE FOR"
1510   MOVE -65,25
1520   LABEL "REMAINING COMPONENTS"
1530   MOVE -65,20
1540   LABEL "CURRENT SOURCES I1=";I1;" I2=";I2
1550   PENUP
1560   PEN 0
1570   Slope=1.E+10    ! VERTICAL VECTOR CASE
1580   IF ABS(ABS(Angle(Worst))-90)>1.0E-4 THEN Slope=TAN(Angle
(Worst))
1590   !
1600                  Nres=13
1610   FOR Res=1 TO Nres  ! NRES=NUMBER OF FAULTY RESISTORS
1620   !
1630   RESTORE R
1640   READ R(*)
1650   R(Worst)=R_value(Res)
1660   CALL Dc_circuit(I1,I2,R(*),V1,V2,Delta_v(*))
1670   V1=V1-Origin(1)   ! DELTA VOLTAGE
1680   V2=V2-Origin(2)   ! DELTA VOLTAGE
1690         Xplot(Res,2)=V1
1700         Yplot(Res,2)=V2
1710   PIVOT 0
1720       FOR Combo=0 TO 255  ! ALL COMBINATIONS OF COMPONENT
+ OR - 5% TOLERANCE
1730       X(Combo)=0
1740       Y(Combo)=0
```

```
1750        Error_ang(Combo)=0
1760            FOR I=0 TO 7
1770            Component=I+4
1780            IF Component=Worst THEN Skip
1790            IF BIT(Combo,I)=0 THEN
1800            X(Combo)=X(Combo)+Origin_low(Component,1)
1810            Y(Combo)=Y(Combo)+Origin_low(Component,2)
1820            Error_ang(Combo)=Error_ang(Combo)+Angle_low(Comp
onent,Worst)-Angle(Worst)
1830            ELSE
1840            X(Combo)=X(Combo)+Origin_high(Component,1)
1850            Y(Combo)=Y(Combo)+Origin_high(Component,2)
1860            Error_ang(Combo)=Error_ang(Combo)+Angle_high(Com
ponent,Worst)-Angle(Worst)
1870            END IF
1880 Skip:      NEXT I
1890        !
1900        ! ROTATE AND TRANSLATE V1 AND V2 BY
1910        ! ERROR ANGLE AND ORIGIN SHIFT
1920        !
1930        X(Combo)=X(Combo)+V1*COS(Error_ang(Combo))-V2*SIN(Er
ror_ang(Combo))
1940        Y(Combo)=Y(Combo)+V1*SIN(Error_ang(Combo))+V2*COS(Er
ror_ang(Combo))
1950        Measure_dist(Combo)=(+Slope*X(Combo)-Y(Combo))/SQR(S
lope*Slope+1)
1960        !
1970        !
1980        NEXT Combo
1990        !
2000        ! FIND MINIMUM AND MAXIMUM
2010        !
2020        Max=-1.E+10
2030        Min=+1.E+10
2040        !
2050          FOR I=0 TO 255
2060              IF Measure_dist(I)>Max THEN
2070              Max=Measure_dist(I)
2080              Imax=I
2090              END IF
2100              IF Measure_dist(I)<Min THEN
2110              Min=Measure_dist(I)
2120              Imin=I
2130              END IF
2140          NEXT I
2150          !
2160          ! SOLVE EXACT CASE
2170          Pattern=0
2180          FOR I=0 TO 7
2190          Pattern=10*Pattern+BIT(Imin,7-I)
2200          IF I+4=Worst THEN Skip2
```

```
2210          IF BIT(Imin,I)=0 THEN R(I+4)=R(I+4)*.95
2220          IF BIT(Imin,I)=1 THEN R(I+4)=R(I+4)*1.05
2230 Skip2:           NEXT I
2240                  CALL Dc_circuit(I1,I2,R(*),V1,V2,Delta_v(*)
)
2250                  X(Imin)=V1-Origin(1)
2260                  Y(Imin)=V2-Origin(2)
2270                  RESTORE R
2280                  READ R(*)
2290                  R(Worst)=R_value(Res)
2300                  Pattern=0
2310          FOR I=0 TO 7
2320          Pattern=10*Pattern+BIT(Imax,7-I)
2330          IF I+4=Worst THEN Skip3
2340          IF BIT(Imax,I)=0 THEN R(I+4)=R(I+4)*.95
2350          IF BIT(Imax,I)=1 THEN R(I+4)=R(I+4)*1.05
2360 Skip3:         NEXT I
2370                CALL Dc_circuit(I1,I2,R(*),V1,V2,Delta_v(*))
2380                X(Imax)=V1-Origin(1)
2390                Y(Imax)=V2-Origin(2)
2400          Xplot(Res,1)=X(Imin)
2410          Yplot(Res,1)=Y(Imin)
2420          Xplot(Res,3)=X(Imax)
2430          Yplot(Res,3)=Y(Imax)
2440   NEXT Res
2450          ! PLOT THE NOMINAL VECTOR AND
2460          ! MIN AND MAX TOLERANCE WORST CASE
2470          ! LINES
2480          LORG 5
2490          PEN 2
2500          FOR I=1 TO 3
2510              MOVE 50*Xplot(1,I)+.25,50*(Yplot(1,I))+.75
2520              LABEL "."
2530              FOR Res=2 TO Nres
2540              MOVE 50*Xplot(Res-1,I),50*Yplot(Res-1,I)
2550              DRAW 50*Xplot(Res,I),50*Yplot(Res,I)
2560              MOVE 50*Xplot(Res,I)+.25,50*Yplot(Res,I)+.75
2570              LABEL "."
2580              NEXT Res
2590          NEXT I
2600   ! DRAW LINES FROM MIN TO MAX WORST CASE
2610   !
2620          FOR Res=1 TO Nres
2630          MOVE 50*Xplot(Res,1),50*Yplot(Res,1)
2640          DRAW 50*Xplot(Res,2),50*Yplot(Res,2)
2650          DRAW 50*Xplot(Res,3),50*Yplot(Res,3)
2660          NEXT Res
2670   !
2680   ! LABEL FAULT RESISTOR VALUE ON PLOT
2690   !
2700          CSIZE 2.5
```

```
2710          IF Angle(Worst)>=0 AND Angle(Worst)<14. THEN LORG
6       !CENTERED
2720          IF Angle(Worst)>=14. AND Angle(Worst)<=76. THEN LO
RG 3  !UPPER LEFT
2730          IF Angle(Worst)>76. THEN LORG 2
2740          IF Angle(Worst)<0 AND Angle(Worst)>-14. THEN LORG
4  !BOTTOM CENTERED
2750          IF Angle(Worst)<=-14. THEN LORG 1  !LOWER LEFT
2760          !
2770          I=3   !I DETERMINES WHICH SIDE OF THE PLOT THAT VA
LUE LABELS ARE PLACED
2780          IF Angle(Worst)<0 AND Angle(Worst)>-89. THEN I=1
2790              MOVE 50*Xplot(1,I),50*Yplot(1,I)
2800              LABEL USING ".DD";R_value(1)
2810              X_last=Xplot(1,I)
2820              Y_last=Yplot(1,I)
2830          FOR Res=2 TO Nres
2840          IF 50*ABS(Xplot(Res,I)-X_last)>1.8*(Digits+2) OR 5
0*ABS(Yplot(Res,I)-Y_last)>1.8 THEN
2850              MOVE 50*Xplot(Res,I),50*Yplot(Res,I)
2860                 Digits=0
2870              IF R_value(Res)<1.0 THEN
2880                 LABEL USING ".DD";R_value(Res)
2890              ELSE
2900                 Digits=1.0+INT(LGT(R_value(Res)))
2910                 LABEL USING VAL$(Digits)&"D.DD";R_value(Res
)
2920              END IF
2930              X_last=Xplot(Res,I)
2940              Y_last=Yplot(Res,I)
2950              END IF
2960          NEXT Res
2970          LDIR 0
2980          PENUP
2990          PEN 0
3000    NEXT Worst  ! NEXT CASE
3010          !
3020    END
```

```
10     SUB Dc_circuit(I1,I2,R(*),V1,V2,Delta_v(*),OPTIONAL Flag
)
20     ! PROGRAM "DC_CIRCUIT"
30     !
40     !
50     ! INPUTS TO THIS SUBROUTINE ARE CURRENT
60     ! SOURCE VALUES I1 AND I2 AND THE RESISTANCE
70     ! ARRAY R(*) FOR THE DC CIRCUIT.
80     !
90     ! OUTPUTS FROM THIS SUBROUTINE ARE PORT
100    ! VOLTAGES V1 AND V2 AND DELTA_V(*) WHICH
110    ! IS THE ARRAY OF VOLTAGE CHANGES INDUCED
120    ! BY THE FAULT CURRENTS
130    !
140    ! THE PRINT FLAG IS AN OPTIONAL FLAG FOF
150    ! PROGRAM DEVELOPMENT PURPOSES.
160    !
170    DIM Q(1:10,1:10),B(1:10,1:10)
180    DIM Asave(1:10,1:10),Identity(1:10,1:10)
190    DIM Tolerance(1:3),Angle(1:3,4:11)
200    !
210    !
220    IF NPAR<7 THEN Print_flag=0
230    IF NPAR=7 THEN Print_flag=Flag ! SET OPTION AL PRINT FLA
G PARAMETER
240    !
250    !
260    RESTORE
270    !
280    !
290    !
300    DATA 1,-1,-1,0,0,0,0,-1,0,0
310    DATA 0,0,1,-1,-1,0,0,0,0,0
320    DATA 0,0,0,0,1,-1,-1,1,0,0
330    DATA 1,0,0,0,0,0,0,0,0,0
340    DATA 0,0,0,0,0,0,1,0,0,0
350    DATA 0,-1.,1,1.0,0,0,0,0,0,0
360    DATA 0,0,0,-1,1,1,0,0,0,0
370    DATA 0,0,-1,0,-1,0,0,.5,0,0
380    DATA 1,1.00,0,0,0,0,0,0,-1,0
390    DATA 0,0,0,0,0,-1,1,0.0,1
400    N=10
410    FOR I=1 TO N
420    FOR J=1 TO N
430    READ Q(I,J)
440    NEXT J
450    NEXT I
460    ! RESISTANCES
470    !
480    Q(9,1)=R(4)
490     Q(6,2)=-R(5)
```

```
500     Q(9,2)=R(5)
510     Q(6,3)=R(6)
520     Q(8,3)=-R(6)
530     Q(6,4)=R(7)
540     Q(7,4)=-R(7)
550     Q(7,5)=R(8)
560     Q(8,5)=-R(8)
570     Q(7,6)=R(9)
580     Q(10,6)=-R(9)
590     Q(10,7)=R(10)
600     Q(8,8)=R(11)
610    !
620    !
630     IF Print_flag=1 THEN PRINT "Q MATRIX -- CONNECTION AND
COMPONENTS"
640    IF Print_flag=1 THEN PRINT
650    FOR I=1 TO N
660     IF Print_flag=1 THEN PRINT
670    FOR J=1 TO N
680    Asave(I,J)=Q(I,J)
690    IF Print_flag=1 THEN PRINT USING "3D.DD,#";Q(I,J)
700    NEXT J
710    IF Print_flag=1 THEN PRINT
720    NEXT I
730    !
740    ! REPLACE MATRIX VALUES BY TRUE RESISTANCES
750    !
760    CALL Matinv(N,Q(*),B(*),Det)
770   ! DC CIRCUIT CASE
780    !
790    ! DEFINE F MATRIX
800    DIM F(1:10,1:10)
810    !
820    DATA 1,-1,-1,0,0,0,0,-1,0,0
830    DATA 0,0,1,-1,-1,0,0,0,0,0
840    DATA 0,0,0,0,1,-1,-1,1,0,0
850    DATA 1,0,0,0,0,0,0,0,1,0
860    DATA 0,0,0,0,0,0,1,0,0,-1
870    DATA 0,0,0,0,0,0,0,0,0,0
880    DATA 0,0,0,0,0,0,0,0,0,0
890    DATA 0,0,0,0,0,0,0,0,0,0
900    DATA 0,0,0,0,0,0,0,0,0,0
910    DATA 0,0,0,0,0,0,0,0,0,0
920    !
930    FOR I=1 TO N
940    FOR J=1 TO N
950    READ F(I,J)
960    NEXT J
970    NEXT I
980    !
990    CALL Matmult(B(*),F(*),Q(*),N,N,N)
```

```
1000  !
1010  DIM M(1:2,1:10)
1020  DATA 0,0,0,0,0,0,0,0,0,0
1030  DATA 0,0,0,0,0,0,0,0,0,0
1040  !
1050  FOR I=1 TO 2
1060  FOR J=1 TO N
1070  READ M(I,J)
1080  NEXT J
1090  NEXT I
1100  !
1110  M(1,1)=R(4)
1120  M(1,2)=R(5)
1130  M(2,6)=R(9)
1140  M(2,7)=-R(10)
1150  !
1160  CALL Matmult(M(*),Q(*),Delta_v(*),2,10,10)
1170  V1=I1*Delta_v(1,9)+I2*Delta_v(1,10)
1180  V2=I1*Delta_v(2,9)+I2*Delta_v(2,10)
1190  IF Print_flag=1 THEN
1200  PRINT
1210  PRINT "DELTA V VECTOR"
1220  PRINT
1230  FOR I=1 TO 2
1240  PRINT
1250  PRINT
1260  FOR J=1 TO N
1270  PRINT USING " 4D.D.XX,#";24*Delta_v(I,J)
1280  NEXT J
1290  NEXT I
1300  !
1310  PRINT
1320  PRINT "FAULT PATTERN ANGLES"
1330  FOR J=1 TO N-2
1340  PRINT USING """G"",DD,XX,""ANGLE="",DDDD.DDD";J+3,FNAtn(
Delta_v(1,J),Delta_v(2,J))
1350  NEXT J
1360  !
1370  PRINT "V1=";V1,"V2=";V2
1380  END IF
1390  !
1400  !
1410  !
1420  !
1430  !
1440  !
1450  SUBEND
```

```
10     SUB Dc_plot(Angle(*),OPTIONAL Vm)
20     ! LIN-LIU FAULT VECTOR PLOT
30     !
40     ! GENERATES THE BACKGROUND
50     ! INFORMATION FOR THE PLOTS
60     !
70     GINIT
80     GRAPHICS ON
90 !!!PLOTTER IS 705,"HPGL"
100    Vmax=1.0
110    IF NPAR=2 THEN Vmax=Vm    ! OPTIONAL SCALE FOR MAX VALUE
OF VOLTAGE
120    PEN 1
130    PENUP
140    FRAME
150    CSIZE 5
160    LORG 5
170    SHOW -50,50,-50,50
180    MOVE -10,40
190    LABEL "V2"
200    MOVE 40,-10
210    LABEL "V1"
220    DEG
230        MOVE -50,0
240        DRAW 50,0
250        MOVE 0,-50
260        DRAW 0,50
270        CLIP -50,50,-50,50
280        AXES 10,10
290        CLIP OFF
300        CSIZE 3
310        LORG 6          ! LABEL X AXIS
320        MOVE -50,0
330        LABEL USING "DD.D";-Vmax
340        MOVE 50,0
350        LABEL USING "DD.D";+Vmax
360        LORG 7          ! LABEL Y AXIS
370        MOVE 0,-50
380        LABEL USING "DD.D";-Vmax
390        LORG 9
400        MOVE 0,50
410        LABEL USING "DD.D";+Vmax
420      LORG 1
430      FOR J=4 TO 11
440        MOVE 45*COS(Angle(J)),45*SIN(Angle(J))
450       IF J>9 THEN LABEL USING """G"",DD";J
460       IF J<10 THEN LABEL USING """G"",D";J
470      NEXT J
480    SUBEND
```

```
10     DEF FNAtn(X,Y)
20     ! TWO QUADRANT ARCTANGENT FUNCTION
30     !
40     DEG
50     IF ABS(X)<1.E-4 THEN RETURN -90
60     RETURN ATN(Y/X)
70     !
80 ! KEEP THIS LINE FOR 4-QUADRANT VERSION   IF X<0 THEN RETUR
N SGN(Y)*180+ATN(Y/X)
90 ! KEEP FOR 4-QUAD       IF X>0 THEN RETURN ATN(Y/X)
100    FNEND
```

```
10      SUB Matmult(A(*).B(*),C(*),N,K,M)
20      !
30      ! SUBROUTINE TO MULTIPLY TWO MATRICES
40      !
50      ! DIMENSIONS ARE N X K  AND K X M
60      !
70            FOR N_index=1 TO N
80                FOR M_index=1 TO M
90                    C(N_index,M_index)=0
100               NEXT M_index
110           NEXT N_index
120     !
130     !
140           FOR N_index=1 TO N
150               FOR M_index=1 TO M
160                   FOR K_index=1 TO K
170                   C(N_index,M_index)=C(N_index,M_index)+A(N_
index,K_index)*B(K_index,M_index)
180                   NEXT K_index
190               NEXT M_index
200           NEXT N_index
210     !
220     SUBEND
```

```
10     SUB Matinv(N,A(*),B(*),Det)
20     !
30     !  INVERTS A REAL N X N MATRIX A AND
40     !  STORES THE RESULT IN B
50     !
60        INTEGER I,J
70        DIM P(1:10)
80     !
90     !
100    Det=FNFactor(N,A(*),P(*))
110    !
120    IF ABS(Det)<1.E-20 THEN
130       ! BOMB CITY !
140       PRINT "DETERMINANT LESS THAN 1E-20 ** NON-INVERTABLE
MATRIX"
150       PRINT "VALUE OF DETERMINANT=",Det
160       RETURN
170    END IF
180    !  SET UP AUGMENTATION MATRIX B
190    !
200       FOR I=1 TO N
210           FOR J=1 TO N
220           B(I,J)=0.
230           IF I=J THEN B(I,J)=1.0
240           NEXT J
250       NEXT I
260    !
270    !
280    CALL Lu(A(*),P(*),N,B(*),N)
290    !
300    SUBEND
310    !
320    !
330    !
340    !  DEFINE FACTOR FUNCTION
350    !
360    DEF FNFactor(N,A(*),P(*))
370    !
380       INTEGER I,J,K,K1,N1,Pk,Ppi
390       !
400       FOR I=1 TO N
410           P(I)=I
420       NEXT I
430       !
440       N1=N-1
450       Delta=1.0
460       !
470       FOR K=1 TO N1
480           Pk=P(K)
490           Max=ABS(A(Pk,K))
500           FOR I=K TO N
```

```
510             Ppi=P(I)
520             IF ABS(A(Ppi,K))<Max THEN
530             ELSE
540             Max=ABS(A(Ppi,K))
550             Pk=I
560             Delta=-Delta
570             END IF
580         NEXT I
590         !
600         IF Max>0 THEN
610             ELSE
620             RETURN 0
630             !
640             END IF
650             !
660         Ppi=P(K)
670         P(K)=P(Pk)
680         P(Pk)=Ppi
690         K1=K+1
700                 FOR I=K1 TO N
710                 Ppi=P(I)
720                 A(Ppi,K)=A(Ppi,K)/A(P(K),K)
730                     FOR J=K1 TO N
740                     A(Ppi,J)=A(Ppi,J)-A(Ppi,K)*A(P(K),J)
750                     NEXT J
760                 NEXT I
770         Delta=Delta*A(P(K),K)
780     NEXT K
790         !
800     RETURN Delta*A(P(N),N)
810         !
820 FNEND
830 !
840 !
850 !
860 !
870       ! SUB PROGRAM  LU
880 !
890 SUB Lu(A(*),P(*),N,B(*),M)
900 !
910     DIM T(1:10)
920     INTEGER K,I,J,Ilim,Ic,H,Ip1
930 !
940     FOR K=1 TO M
950         T(1)=B(P(1),K)
960             FOR I=2 TO N
970                 H=P(I)
980                 T(I)=B(H,K)
990                 Ilim=I-1
1000                    FOR J=1 TO Ilim
1010                        T(I)=T(I)-A(H,J)*T(J)
```

```
1020                        NEXT J
1030                    NEXT I
1040                T(N)=T(N)/A(P(N),N)
1050                B(N,K)=T(N)
1060                    FOR I=2 TO N
1070                        Ic=N+1-I
1080                        H=P(Ic)
1090                        Ip1=Ic+1
1100                            FOR J=Ip1 TO N
1110                               T(Ic)=T(Ic)-A(H,J)*T(J)
1120                            NEXT J
1130                        T(Ic)=T(Ic)/A(H,Ic)
1140                        B(Ic,K)=T(Ic)
1150                    NEXT I
1160        NEXT K
1170    SUBEND
1180    !
1190    !
1200    !
```

END

FILMED

3-84

DTIC